mathematics

MDPI

Article

DeepBlockShield: Blockchain Agent-Based Secured Clinical Data Management Model from the Deep Web Environment

Junho Kim and Mucheol Kim *

Department of Computer Science and Engineering Chung-Ang University, Seoul 06974, Korea; kjhcau@gmail.com
* Correspondence: kimm@sw.cau.ac.kr or mucheol.kim@gmail.com

Abstract: With the growth of artificial intelligence in healthcare and biomedical research, many researchers are interested in large amounts of data in hospitals and medical research centers. Then the need for remote medicine services and clinical data utilization are expanding. However, since the misuse and abuse of clinical data causes serious problems, the scope of its use is bound to have a limited range physically and logically. Then a security-enhanced data distribution system for medical deep web environments. Therefore, in this paper, we propose a blockchain-based clinical data management model named DeepBlockshield to prevent information leakage between the deep web and the surface web. Blockchain supports data integrity and user validation to support data sharing in closed networks. Meanwhile, the agent performs integrity verification between the blockchain and the deep web and strengthens the security between the surface web and the deep web. DeepBlockShield verifies the user's validity through the records of the deep web and blockchain. Furthermore, we wrap the results analyzed by the valid request into a web interface and provide information to the requester asynchronously. In the experiment, the block generation cycle and size on the delay time was analyzed for verifying the stability of the blockchain network. As a result, it showed that the proposed approach guarantees the integrity and availability of clinical data in the deep web environment.

Keywords: blockchain; healthcare; clinical data; security; deep web

Citation: Kim, J.; Kim, M. DeepBlockShield: Blockchain Agent-Based Secured Clinical Data Management Model from the Deep Web Environment. *Mathematics* **2021**, *9*, 1069. https://doi.org/10.3390/math9091069

Academic Editor: Damien Sauveron

Received: 1 March 2021
Accepted: 5 May 2021
Published: 10 May 2021

1. Introduction

In spite of the fact that the development of artificial intelligence (AI) in healthcare and biomedical research is advancing rapidly, there are many challenges and limitations (i.e., privacy, diagnosis ethics) in the real-world use of AI algorithms in clinical practice. Advanced bioinformatics for diagnosis, prognosis, and treatment can be realized based on a variety of data, then a system for secured clinical data distribution should be ensured. Meanwhile, information produced in various industries is being loaded into a closed network, and while actively participating in the production process as well as big data and IoT, the web becomes more diverse and subdivided [1–3].

Figure 1 shows the hierarchy of web environments which consist of surface, deep and dark web. Surface Web generally means World Wide Web (WWW), namely many social network services (i.e., Facebook, Twitter) and search engines (i.e., Google, Yahoo, and Bing) belong to it. The deep web, separated from the surface web, has accumulated over 90% of data in various fields such as language, medicine, and government [4–6]. It takes the role of a specialized data warehouse in a closed network environment. Hence, it is not only inaccessible to unauthorized people; it also does not allow indexing in regular search engines.

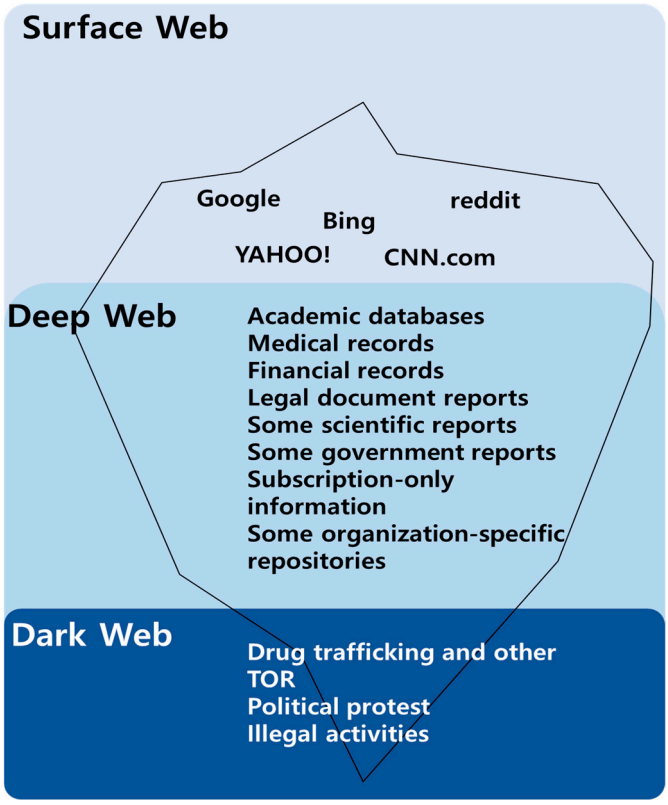

Figure 1. The concept of surface, deep and dark web.

Typically, clinical data resides on the deep web. In fact, it exists on the deep web and is provided in a limited format to prevent the leakage of sensitive personal information. However, the use of this data is increasing for the development of diagnostic technology and services [7]. In addition, the scope of use for medical disputes and criminal investigations has expanded even further. Its use was generally not permitted due to the potential for certain problems, including unauthorized use, leakage, abuse and misuse.

Currently, information-sharing programs in the field of medical research focus solely on collection and use. Most of the data generated by medical institutions is stored in databases. It is basically encrypted using digital signatures or DRM (digital rights management). However, in this storage method, if someone obtains the authority of an administrator, clinical data can be edited and arbitrarily modified. It can also be forged by an authorized person while being transferred or stored in the database. Moreover, if it is transmitted in text format on the network, it can be leaked by sniffing attack [8]. Recently, research on strengthening security based on blockchain is underway. In general, in order to safely utilize this technology, the stability of the technology is secured by establishing a network through a private blockchain and consensus algorithm in which only authorized users participate [9–12]. In the case of shared ledgers, the history is recorded, and the usage flow is verified [13,14].

With the innovation of artificial intelligence technology, the use of clinical data in a deep web environment has become important. However, due to personal information leakage and abuse, its use still remains stagnant. In this paper, we implement a blockchain-based agent that prevents information leakage and provides a secure clinical data management model. The proposed DeepBlockShield implements a horizontal security system that prevents data leakage in the surface web environment and a vertical management system between the deep web and the surface web. The agent performs authentication to determine access rights to clinical data, and users can be divided to researchers and administrators according to their rights. Data analysis requests are passed through the agent, then the request and response for clinical data analysis are recorded

into the blockchain. It enhances security by wrapping it asynchronously through a web interface.

This paper is structured as follows: in Section 2, previous studies are reviewed. In Section 3, the architecture design of the proposed system is explained. In Section 4, experiments are analyzed. In Section 5, conclusions are given.

2. Related Work

In general, the blockchain model ensures transaction transparency by allowing anyone to access data. Therefore, there is a risk of leakage when using clinical data through blockchain. In addition, data is anonymized through hash encryption. However, there is a risk of information leakage due to the transparency and re-identification of the data. Research is underway on access control to address the data leaked problem and anonymization approaches to protect information in the process of data disclosure and utilization. Table 1 shows a summary of the work involved.

Table 1. Related Work Summary.

Purpose	List	Summary
Privacy Protection	[12,15–19]	Personal data management platform focused on personal information protection
Access Control	[18–23]	Address data protection and privacy access control issues
Blockchain Architecture	[15,17,18,24,25]	Guaranteed data integrity and non-repudiation through public verification based on blockchain technology

First, various studies were conducted on the definition and importance of deep web data. Bergman et al. [5] quantified the size and importance of deep web content and extracted quality and relevance to search users. They defined the deep web and suggested how to navigate deep web content in search engines. Hu, Vincent C. et al. [7] proposed an access control model by evaluating the rules for the environment related to the properties of subjects and objects, tasks and requests. It supports a flexible approach to implementing access control policies that are limited only to the computational language and the abundance of available attributes.

In order to solve the problem of data access to personal information, Zyskind et al. [16] conducted a study on the control and management of user data through agents and the storage and access control of data using blockchain blocks [23]. Maymounkov et al. [21] utilized distributed hash tables to solve data protection and privacy access control issues [22]. Kaaniche et al. [20] proposed a method of enhancing security through encryption/signing of data to be shared by encryption based on hierarchical IDs to protect privacy and ensure continuous data availability. Liang et al. [24] proposed a blockchain-based data source architecture to provide tamper-resistant records, enabling transparency of data responsibility in the cloud, and improving the privacy and availability of source data. Rantos et al. [25] proposed a blockchain-based consent management system in which data subjects can exercise their rights for data processing with the goal of meeting the General Data Protection Regulation (GDPR) requirements for personal information protection in IoT environments. Fatokun et al. [17] proposed a patient-centric application that allows patients to manage medical records using blockchain. It enhances security including privacy protection and reinforces interoperability to support data exchange between different healthcare providers. Jabarulla et al. [18] proposed a decentralized patient-centered image management approach as an efficient data sharing of medical big data in an unreliable environment using blockchain and IPFS (Inter Planetary File System) technology. Wehbe et al. [19] proposed a safe management and efficient data integration model using blockchain and AI for the Electronic Healthcare Records (EHR) platform. It effectively tracks the use of patient data and securely maintains data provider ownership through patient-centric access control.

In addition, Mercer [12] proposed a cryptographic protocol designed not to allow individuals to be specified by mixing public keys that can be accessed without identifying the identity of private signers to address the anonymity problem of the blockchain. Kosba [15] proposes a decentralized smart contract system that maintains transaction privacy from the public's point of view without transparently storing transactions to protect transaction privacy.

3. Blockchain Agent Based Secured Clinical Data Management Model

In this subsection, we describe a data management model that can prevent information leakage while expanding the availability of clinical data in the deep web environment. This paper proposes a blockchain-based DeepBlockShield model that implements secure sharing of clinical data (Figure 2). The proposed model adopts a two-way user verification and asynchronous information provision methodology to enhance the security of clinical data.

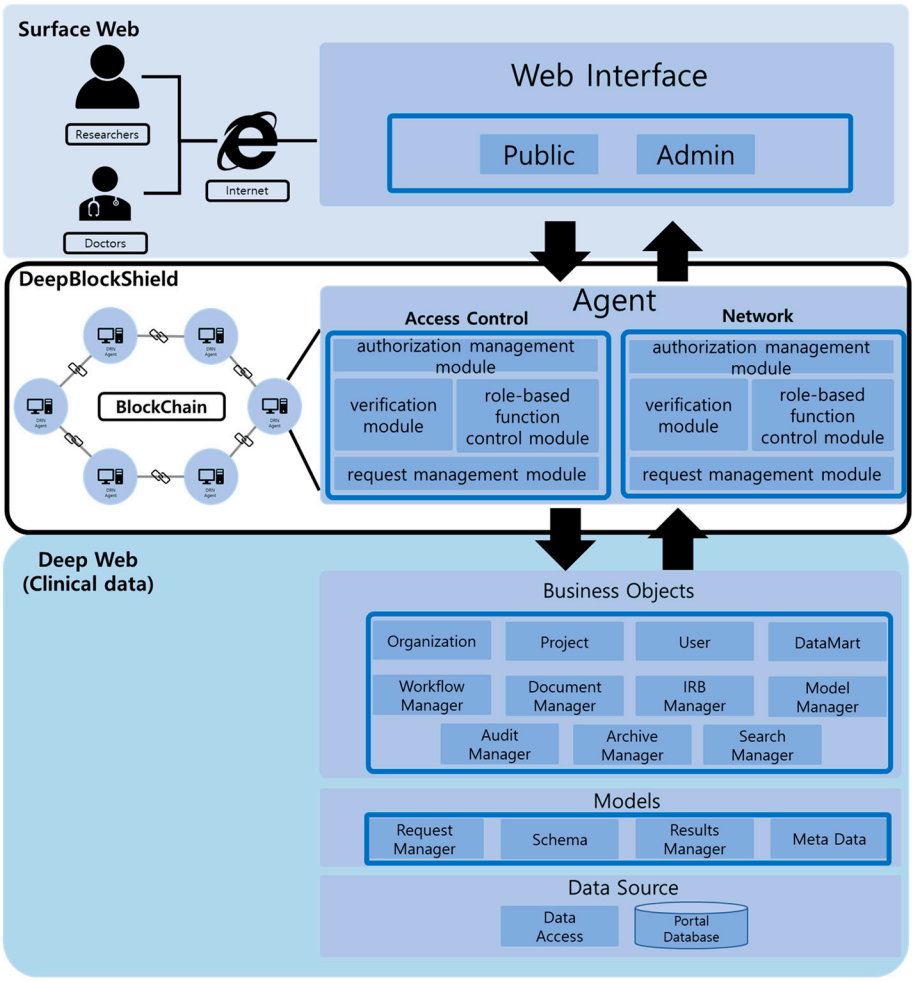

Figure 2. DeepBlockShield Overview.

DeepBlockShield is functionally composed of a verification module, authorization management module, request management module, and role-based function control module. First, the verification module verifies the validity of user access with records between agent and blockchain. Second, the authorization module classifies researchers/administrators according to the agent's permission. Third, the request management module records the request information and analysis results in the blockchain through the agent and provides the asynchronously wrapped outcomes through the interface. Fourth, the role-based function control module approves/rejects analysis request contracts and requests and configures contract items that confirm the results of medical information.

3.1. Separated Access Control Module for Data Security

The proposed model constructs a data-separated agent between the surface web and the deep web as shown in Figure 3 above for the security of clinical data. For the individual management of personal data, blockchain is implemented in an off-chain state. The agent grants a permission to use data through permission control for the improvement of data security. A user is able to request access permission as an agent, using the interface. For this, he is required to provide information needed for user validation such as name, department and email address. Then, once the user information is verified, the administrator allocates the access permission to the agent and generates a smart contract on the blockchain, entering permitted user information. The proposed approach edits and prepares a block structure to manage such permission and records. The block structure according to the proposed scheme is shown in Table 2 below.

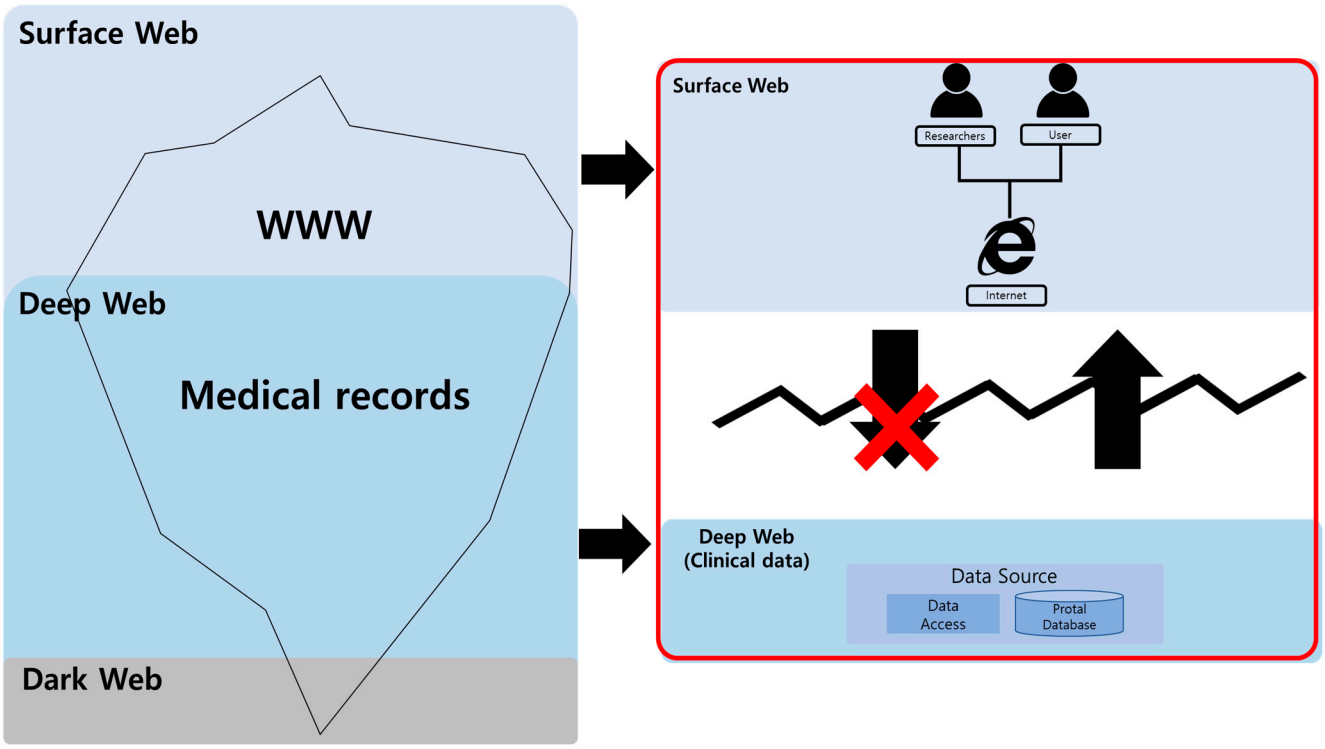

Figure 3. Separated access control module for data security.

Table 2. Block Structure of the Proposed Schema.

Category	Information	Description
Common	Previous Hash	The hash value of the block positioned right before the blockchain.
	Time Stamp	Stores data at the time when a contract is generated, and access permission is approved.
	Research ID	Allocates a unique ID on the researcher joining a blockchain network.
Researcher	Research Period	Stores a period on the researcher's request information.
	Target clinical data	Saves the diagnosis code.
	Variable	Saves a data variable name requested by the researcher based on the CDM catalogue data.
	Free-Text	Stores information on analysis methods by supporting a composite data type, e.g., data analysis request code, other information being generated at analysis request.
Admin	Approval Authority	Allocates a unique ID on the hospitals joining a blockchain network. Expresses information in the institutions holding information on data field names. Allocates institution IDs in sequence when joining a network.
	Analysis Results	A URL revealing analysis results. An access to the site prohibited after an elapse of certain time even though information is leaked.

3.2. Authorization Management Module for Role-Based Smart Contract Configuration

The agent grants right for usage through the authorization management module. It is granted in a differentiated fashion depending on the sharing of clinical data. An agent checks the user's differentiated information permission and automatically generates transactions while creating a smart contract. In other words, smart contracts are managed according to the roles granted through the agent. In addition, it is able to mediate transactions and store and verify all records. As shown in the Figure 4, a blockchain platform builds a data-separated network for an agent-based security management system through the authorization management module. Modules are divided into roles: researcher, admin, and data scientist, and grant authority.

1. Researcher; The researcher belongs to the hospital. Through the agent, clinical data analysis can be requested based on CDM catalog data. Furthermore, by entering the data analysis request code, the researcher enters information on the analysis method desired by the researcher. Finally, the researcher enters the study period for how long the data will be used.
2. Admin; The admin manages the network and assigns a unique ID to hospitals that have joined the blockchain network through an agent. Research can be approved, modified, or canceled based on the information requested by the researcher. In addition, the information analyzed by the data scientist is finally confirmed and the analysis results are provided to the user through the agent.
3. Data scientist; Data scientists have direct access to medical data and check and analyze the data the researcher wants based on the research request information approved by the admin.

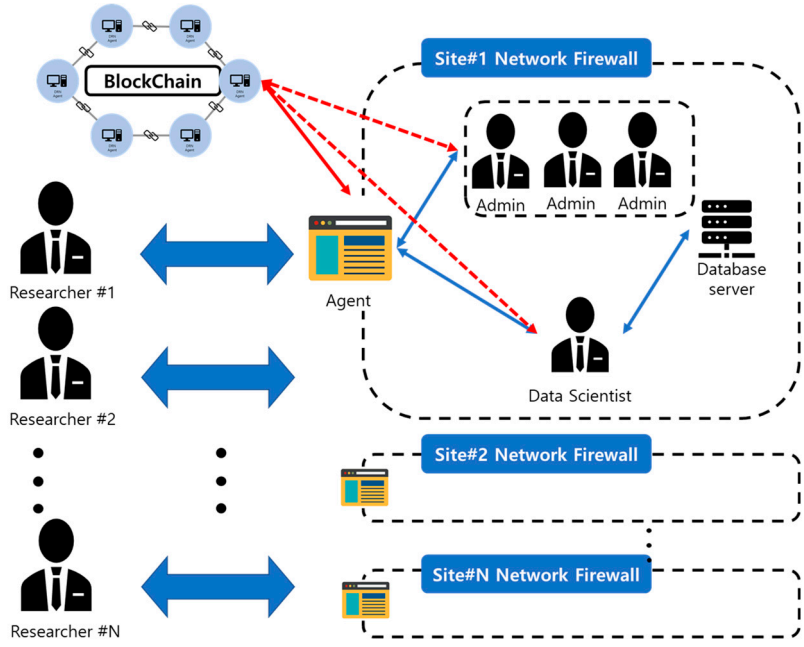

Figure 4. Agent-based security management system and separate blockchain network.

Under such an approach, approved nodes are only allowed to join a network. A blockchain consists of a 'validator node' and 'user node'. In the former, a network is managed through consensus.

Table 3 below states the attributes of authority nodes. 'A_ID' is issued when joining a network and used in checking the creation of a transaction. A ledger is used when all nodes share the same information to check data. The database includes an ID and validation key which are needed to gain access to the external database. The authorization node is given to manage personal information on smart contracts.

Table 3. Attributes of Authority Nodes.

Category	Description
A_ID	Issued through authentication when first joining the network. Block creation and authentication permission. Creates a transaction using A_ID.
Ledger	All authorities share the same contents. History of smart contracts related to the use of personal information by all nodes.
Database	Stores information for database connection. ID and authentication key to access DB provided by institutions
Smart Contract	The management and provision of personal information through smart contract Creates new smart contracts as needed

A user node is a researcher node requesting medical information. Table 4 explains user node attributes. 'U_ID' is issued when joining a network and used in executing a transaction. 'R.I' includes the information on the researchers in Table 2 requested by the user. In addition, data analysis request codes and other information generated at analysis request are entered. A smart contract is a computer protocol designed to register, edit, read and delete medical information.

Table 4. Attributes of User Nodes.

Category	Description
U_ID	Issued through personal information authentication when joining the network Blockchain registration and identifier role when sending personal information. Signing with U_ID when requesting smart contract privacy.
R.I	The R.I contains the researcher information in Table 2 requested by the user Enters the variable name requested by using the variable used. Data analysis request code, other information generated when requesting analysis
Smart Contract	The ability to manage personal information Registration, correction, reading, deletion

Figure 5 reveals a flow of a user's joining of node and authorization. Users register their personal information and affiliation. Once they are verified, they join a network. Then, they can access additional services through an agent.

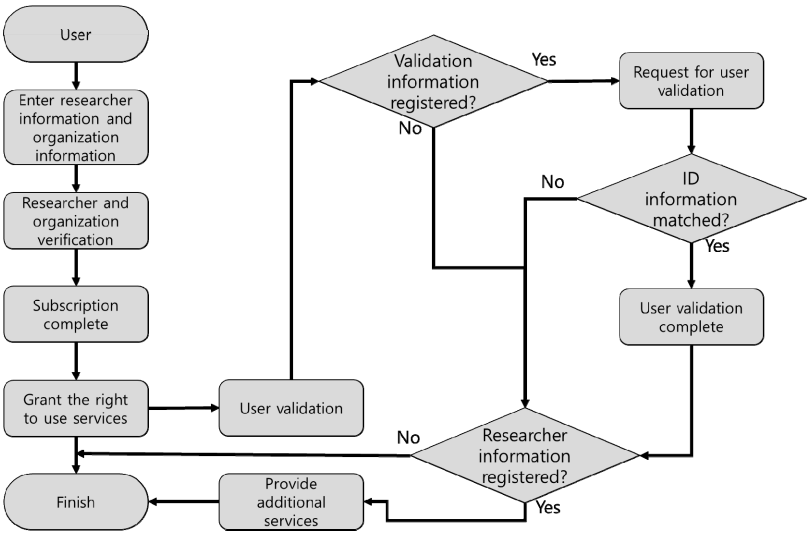

Figure 5. User's node subscription and approval flow.

'U_ID' is used to gain access to services and set smart contract requests. Users check subscription information through the agent and their access permission by confirming service rights. The agent verifies user information registered on the blockchain through the user information stored in the database and 'U_ID' permits access and provides additional services. Such validation information is entered into the blockchain services through the agent. In addition, integrity and effectiveness are validated with hash values between nodes by requesting the utilization of information. The validated data are confirmed with the analysis results whose validation has been proven with the requester's personal key. The ADMIN's registration is also processed just as with the user validation.

3.3. Management of Smart Contract-Based Clinical Data Analysis Request and Role-Based Feature Control

To support the utilization of clinical data and records management, an agent uses a smart contract-based analysis request management module and a role-based feature control module. In the analysis request management module, analysis is requested as illustrated in the Figure 6. According to the figure, once a user logs in through a web interface, user validation is executed through the agent and blockchain. Once the validation is complete, the validation details are recorded through a smart contract. The validated user requests analysis on clinical information. The information requested through the web interface is transmitted to an agent in a JSON format. Then, the agent performs validation to check if the target information exists in the medical database. Once such validation is complete, the details are recorded through the blockchain. After that, if data exist, they are analyzed through a data scientist as shown in Figure 3. The blockchain records the institution which performed such analysis and analysis time and results through a contract. The blockchain encrypts the results and sends them to the agent. Then, the agent provides wrapped results visualized in an asynchronous manner through the VIEW LAYER, using a web interface. The information is readable with the user ID, and the analysis results are protected from a third party. The pseudo-code (Algorithm 1) for user verification is as follows:

Algorithm 1. Pseudo-Code of User Validation

Input: Previous Hash, Time Stamp, User Information
Output: Granting Admin or Research authority depending on whether or not to subscribe to the service

```
U = userType
A = AdminAccount
B = ResearcherAccount
function getUserExist(U) public constant returns(bool)
    if U is Admin, then return A
      if A is not Null, then exist is true
        else if U is Researcher, return B
          if B is not Null, then exist is true
    return exist;
```

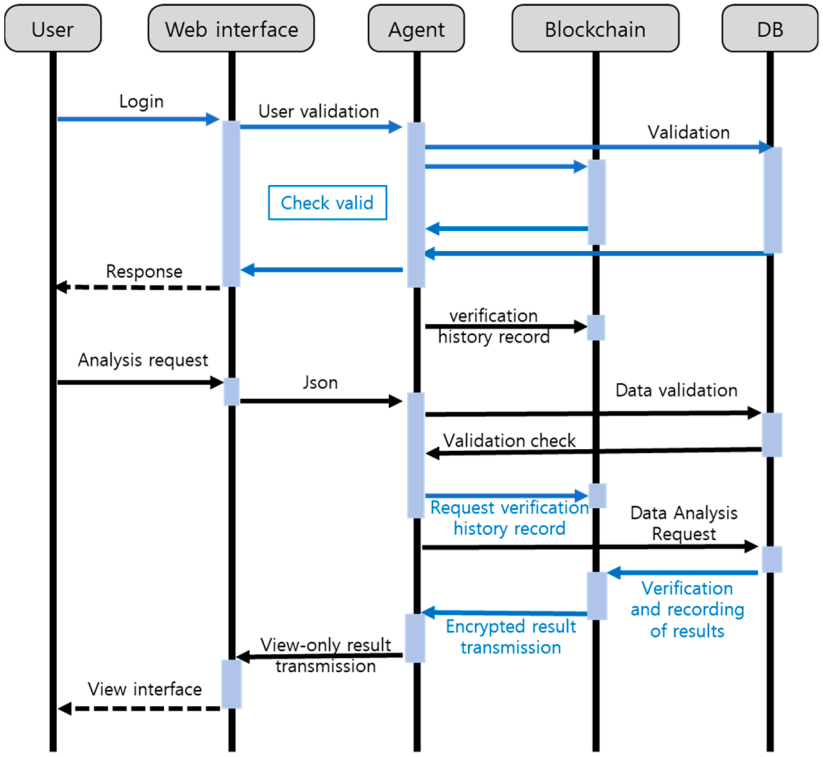

Figure 6. Analysis request management process diagram.

The role-based function control module can create a smart contract through the permission granted through the permission management module. The contract scenario is shown in Figure 7 below.

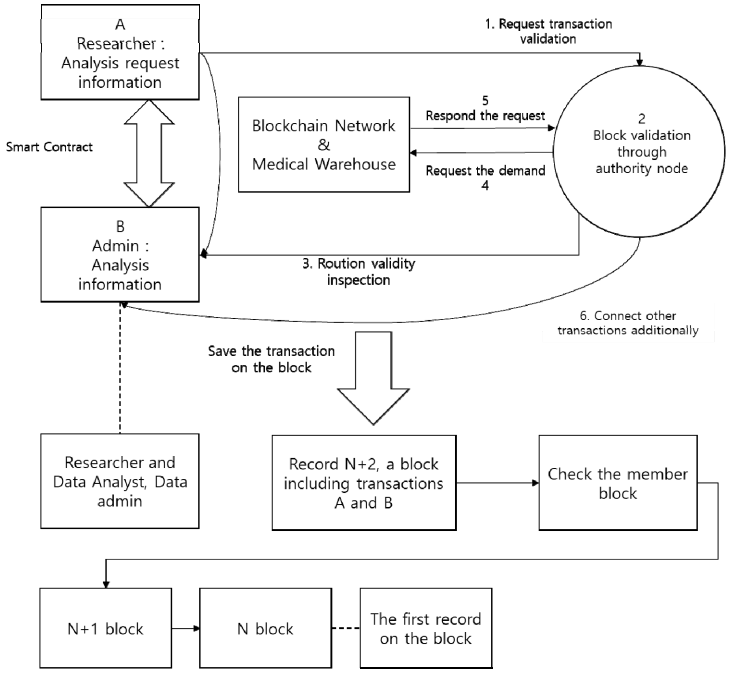

Figure 7. Smart contract scenario diagram.

The information on analysis request is transmitted to the authority node. The authority node verifies the request information records of the user node. For the utilization of validated data, a transaction from the authority node is requested and validated. Once a

user agrees to the use of data, the information requested by the group node is analyzed. The provided results can be checked by the requester's private key. The validator node manages the access to a network by the user node and group node. It handles all transactions on the network and stores and validates transaction history, guaranteeing integrity. A smart contract used by the user node holds request and editing contracts. Information can be requested through each contract.

Table 5 shows the user contract information. In terms of the registration of user contracts, contracts are implemented, using 'U_ID', time stamp, clinical data catalogue and organizational information. In a validator node, utilization is requested by the user node according to the catalogue. In terms of editing, the catalogue information registered by the user upon request is edited by executing a contract. Concerning deletion, request-related records and registration information are deleted.

Table 5. User Contract Configuration.

Category	Description	Input	Output
Request	Registers clinical data to be requested	U_ID, time stamp, Catalog, organization information, Researcher information	Request information
View	Views requested information	U_ID, time stamp	Catalog information requested
Modify	Information correction	U_ID, time stamp, Catalog	Modified information
Cancellation	Cancels analysis request	U_ID, time stamp	Canceled history information

Table 6 shows the information of the Admin contract. In terms of the registration of ADMIN contracts, contracts are implemented, using 'A_ID', time stamp, approved group information and requested clinical data. The group node requests the utilization of personal information through the registered data. In terms of information request, after confirming consent with information from the ADMIN through an agent, a data analyst analyzes the information. Using hash values of the data received after requesting 'A_ID' and results, results can be verified.

Table 6. Admin Contract Configuration.

Category	Description	Input	Output
Approve	Approval of analysis request	A_ID, time stamp, Catalog, organization information, Researcher information	Requested analysis result information
Delete	Deleted requested information	A_ID, time stamp	Deleted history information
Re-request	Cancels analysis request	A_ID, time stamp	Catalog data re-request information

4. Experimental Results

In this section, we implement and experiment the proposed agent-based clinical data management model.

4.1. Experiment Setup

In this paper, a private blockchain was constructed based on the PoA consensus algorithm. The correlation between block generation cycle and size and delay time was analyzed to verify stability. In order to perform the experiment, we (1) adjust the number of authority nodes and the ratio of malicious transactions, and (2) compare the delay time by block size. The agent configuration for the experiment is DB: PostgreSQL, Nodejs: 10.16.3,

OS: Ubuntu 16.04, and the block chain configuration is Go: Go1.12.9, Geth: 1.9.3-stable, RAM: 8 GB, OS: Ubuntu 16.04. Table 7 shows the development environment used to implement the model.

Table 7. Experiment Implementation Environment.

Preferences	Content
OS	Ubuntu 16.04
GO	Go1.12.9
RAM	8GB
Geth	1.9.3-stable
DB	PostgreSQL
Node.js	10.16.3

4.2. Contract Execution

Figure 8 below shows the researcher's interface page for clinical data analysis.

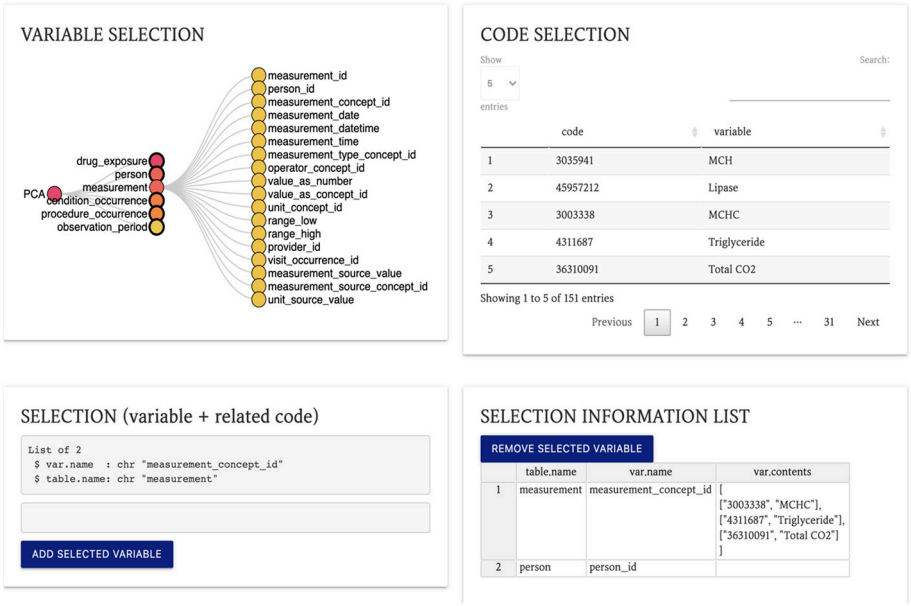

Figure 8. Researcher's interface page.

Both users and admin are able to enter and check the information through the web interface. It sends the information on each process to the blockchain in a JSON format, using a REST method, and a contract is automatically created. In terms of research approval, the encrypted addresses of basic information on the research and user information are stored in blocks. Once the admin approves the analysis request, the query needed for analysis is requested to the user. Then, the user enters the desired catalogue and query through the web interface. The information is entered into blocks through the query request. Then, contract changes can be checked according to individual results. However, if attempting to cancel the research, all details will be deleted. Analysis request information and time records will only remain.

4.3. Blockchain-Based Information Distribution Management Delay Time Analysis

In terms of model performance assessment, a quantitative assessment on the time delayed by the time of creation and block-generation time by the block size and number of nodes is performed. In addition, qualitative assessment on confidentiality, availability and non-repudiation is conducted through a security review.

In our experiments, we measured the delay time by block-generation and TPS. Delay time refers to the time taken for a request to be responded to while TPS represents the number of transactions per second. TPS is calculated by dividing the number of transactions included in the latest block by the block creation cycle. We created 300 transactions per second to measure the model's performance. A block generation interval means the time taken for a new block to be generated. Table 8 shows the block processing time with TPS and latency.

Table 8. TPS and Latency Variation According to Generation Cycle.

Block Generation Cycle (ms)	50	100	250	500	1000	2000
TPS	289	294	275	288	284	291
Latency(sec)	0.30	0331	0.420	0.612	0.802	0.89

As a result of the performance measurement, the longer the block generation cycle, the longer the delay time. However, there is no significant difference in TPS throughput. The block generation cycle is a more important factor than TPS when there are not many transactions. This is because the transaction speed is affected by the block generation cycle. In other words, 50ms is the best block generation cycle.

As a result of measuring network latency by block size, network latency increases as the size increases, while scalability improves. Therefore, the block size is set in consideration of delay time and scalability. Table 9 shows the delay time for each block size.

Table 9. Network Latency by Block Size.

Block Size (MB)	Network Delay (s)
0.1	2
0.5	6
1	11
4	59
8	108

In the detailed options of the blockchain, the proposed model has an average block generation time of 5 s and a transaction processing rate of 300 tps per second. In terms of a block size, if a transaction is requested in the user/validator node, the requested data is only searched. The output values received includes the statistics or results of the application. Then, VIEW through which information can be read is provided through the web interface. In fact, clinical data and analysis results are stored in the database outside the blockchain. Therefore, network speed is kept constant by setting the block size of a model to '1M' in consideration of a rapid increase in the number of transactions due to a low scaling issue for the block size.

Table 10 shows the specific options for each block. The block generation cycle is the time until the next block is generated.

Table 10. Detailed Options for Each Blockchain.

	Consensus	TPS	Block Generation Cycle	Confirmation Frequency	Confirmation Time (sec)
EOS	DPoS	1,000,000	3	15	45
Bitshare	DPoS	100,000	3	15	45
Neo	dBFT	10,000	15–20	1	15–20
Ethereum	PoW	15	14	12	180
Bitcoin	PoW	7	600	6	3600
Proposed	PoA	300	5	2	11

This paper investigates network delay time by block generation interval/block size and finds an optimum size to generate an appropriate network. In conventional blockchain, when the block generation interval was short, data validation time decreased. Therefore, total validation time drops, having a negative effect on reliability. In the proposed model, however, validation reliability is related to the number of authority nodes. In other words, if the number of validation nodes rises, the data validation time also increases. According to the proposed model, a validator chosen based on identity trust, is able to block malicious attacks through block generation, editing and validation. Such a validator proposes and generates blocks. The generated blocks are verified between validators to check data integrity. The Figure 9 below reveals a model comprising three validation nodes and a model with eight different validation nodes. It also compares differences in the time spent to create 1000 blocks. The block size was set to 0.5 M and 1 M.

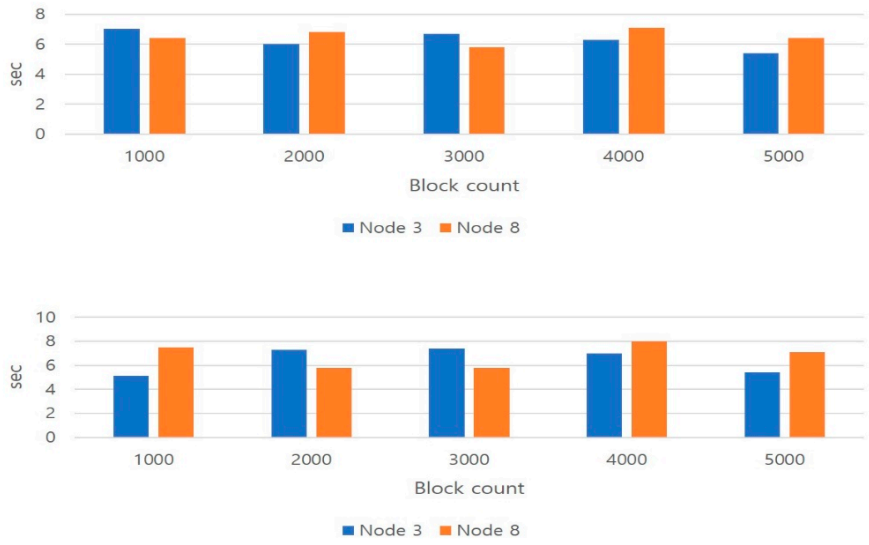

Figure 9. Block Creation Time According to Block Size and Number of Nodes.

Figure 9 counts the generation of thousands of blocks and displays the time the contract was requested and completed. It was created within 9 s on both 0.5 M and 1 M blocks. When eight nodes are configured, as with conventional three-node configuration, only three nodes are selected for validity testing. Therefore, there were no changes in overall performances. In a three-node model, however, when an error occurs in two nodes, a network is shut down. In an eight-node model, on the contrary, even though an error is found in four nodes, the network functions normally.

Figure 10 compares the performance of models using different consensus algorithms under the same conditions (three nodes, block size 1 M). In addition, it shows the block generation time according to the consensus algorithm. The proof of authority (PoA) consensus algorithm is a model optimized for a private net. Since consensus is reached by few authority nodes, it is far faster than the proof of work (PoW) consensus in terms of transaction processing and block generation. In addition, the block generation time is kept almost constant in the PoA algorithm. In the PoW algorithm, on the contrary, as the number of blocks rises, processing time considerably increases.

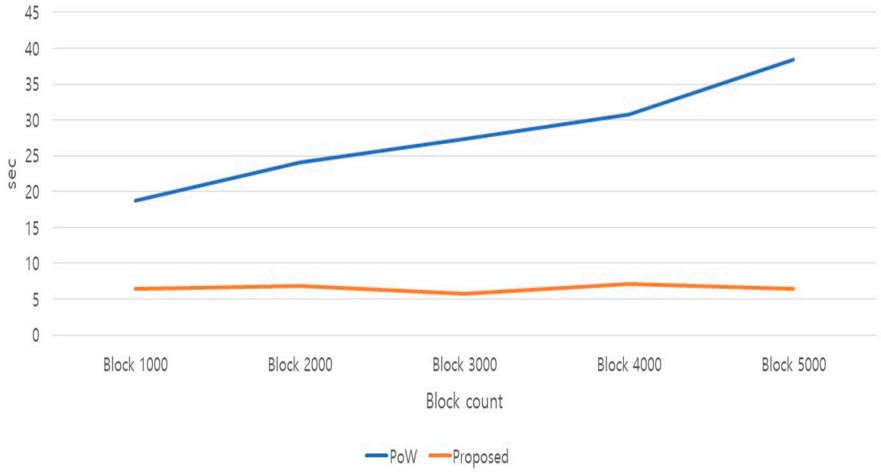

Figure 10. Block generation time according to consensus algorithm.

4.4. Discussions for Security Review with Blockchain Agent

A leak of clinical data can evolve into a privacy breach. Therefore, confidentiality, availability and non-repudiation are reviewed on the features, which are important in terms of stability such as information access permission settings and encryption of shared information.

1. Confidentiality In authority nodes, information is requested to the database configured in an off-chain format, using the data keys requested through user validation. Then, the results are verified with the ID given by the authority node. Then, the encrypted analysis results of the requested data can be checked through the web interface. If data are hacked, encrypted results are only leaked. Unless the encrypted private key granted through the agent is available, analysis results cannot be read. In addition, they are deleted from the database after the elapse of a certain period of time, enhancing confidentiality.
2. Availability The proposed model stores user information in the agent and blockchain as well as through the web interface, allowing authorized members only to access it. In fact, an unauthorized user is not permitted to join the agent and blockchain. In the proposed model, information sharing on a closed network is performed through the blockchain. In the blockchain, validation nodes in which identity is guaranteed through the PoA consensus algorithm are only authorized to create and validate blocks. Even if malicious users attack the services, data cannot be forged or modified because they have no authority of data validation. Therefore, actual network damage is minor. In all nodes, furthermore, authorized validators are only able to validate and extend transactions in a quick and easy fashion.
3. Non-repudiation In the proposed model, concerning the use of clinical data, related information (who, when, what, how) is stored in blocks when a transaction is requested. Then, a validator checks the transaction, and the transaction information is shared through the ledger owned by the network members. In other words, the history of access to personal information is clarified, allowing the information provider, hospitals, or governmental institutes to check the details.

5. Conclusions

With the development of artificial intelligence, remote medical services and research collaboration with common data are activated. However, the scope of data utilization is limited due to the leakage and abuse of personal sensitive information in clinical data. In this paper, we propose a blockchain-based clinical data management model called DeepBlockshield for preventing information leakage between the deep web and the surface web. The proposed model performs validation of user access and its authorization. Moreover, it can

safely provide the analytic results between researchers. It is based on asynchronous data provision and indirect information sharing through the interaction between the blockchain and the agent. As a result, it not only prevents information leakage of medical records, but also innovatively improves interoperability and accessibility in the clinical research data.

The proposed model is designed to be applied to organizations, companies and hospitals where information protection is necessary. It is important to continuously improve the blockchain and data management on closed networks. In addition, further research is necessary to establish the distributed medical research networks based on data standardization. In future research, we intend to further study techniques to defend against attacks of agents.

Author Contributions: Methodology: J.K.; Supervision, M.K.; Writing—Original draft, J.K.; Writing—Review and editing, M.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported in part by a grant of the Korea Health Technology R&D Project through the Korea Health Industry Development Institute (KHIDI), funded by the Ministry of Health & Welfare, Republic of Korea (grant number: HI19C0870) and part by Korea Institute for Advancement of Technology (KIAT) grant funded by the Korea Government (MOTIE) (P0012724, The Competency Development Program for Industry Specialist).

Institutional Review Board Statement: This study performed in accordance with the Declaration of Helsinki and approved by the Institutional Review Board of Catholic University (IRB number: KC19RNSI0624, Approval date: 20190930).

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bilogrevic, I. A machine-learning based approach to privacy-aware information-sharing in mobile social networks. *Pervasive Mob. Comput.* **2016**, *25*, 125–142. [CrossRef]
2. Kim, M.; Sang, O.P. Group affinity based social trust model for an intelligent movie recommender system. *Multimed. Tools Appl.* **2013**, *64*, 505–516. [CrossRef]
3. Kim, M. Identity management-based social trust model for mediating information sharing and privacy enhancement. *Secur. Commun. Netw.* **2012**, *5*, 887–897. [CrossRef]
4. Alba, A. Accessing the deep web: When good ideas go bad. In Proceedings of the Companion to the 23rd ACM SIGPLAN Conference on Object-Oriented Programming Systems Languages and Applications, Nashville, TN, USA, 19–23 October 2008; pp. 815–818.
5. Bergman, M.K. White paper: The deep web: Surfacing hidden value. *J. Electron. Publ.* **2001**, *7*. [CrossRef]
6. He, B. Accessing the deep web. *Commun. ACM* **2007**, *50*, 94–101. [CrossRef]
7. Hu, V.C. Attribute-based access control. *Computer* **2015**, *48*, 85–88. [CrossRef]
8. Kim, Y.-Y.; Seung-soo, S. A study on reliable electronic medical record systems. *J. Digit. Converg.* **2012**, *10*, 193–200.
9. Barcelo, J. User Privacy in the Public Bitcoin Blockchain. 2014. Available online: http://www.dtic.upf.edu/jbarcelo/papers/20140704UserPrivacyinthePublicBitcoinBlockchain/paper.pdf (accessed on 9 May 2016).
10. De Angelis, S. Assessing security and performances of consensus algorithms for permissioned blockchains. *arXiv* **2018**, arXiv:1805.03490.
11. Eyal, I. Bitcoin-ng: A scalable blockchain protocol. In Proceedings of the 13th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 16), Santa Clara, CA, USA, 16–18 March 2016.
12. Mercer, R. Privacy on the blockchain: Unique ring signatures. *arXiv* **2016**, arXiv:1612.01188.
13. Buterin, V. A Next-Generation Smart Contract and Decentralized Application Platform. White Paper 3.37 (2014). Available online: https://ethereum.org/en/whitepaper/ (accessed on 12 November 2020).
14. Guadamuz, A.; Christopher, M. Blockchains and Bitcoin: Regulatory responses to cryptocurrencies. *First Monday* **2015**, *20*. [CrossRef]
15. Kosba, A. Hawk: The blockchain model of cryptography and privacy-preserving smart contracts. In Proceedings of the 2016 IEEE Symposium on Security and Privacy (SP), San Jose, CA, USA, 22–26 May 2016; pp. 839–858.
16. Zyskind, G.; Oz, N. Decentralizing privacy: Using blockchain to protect personal data. In Proceedings of the 2015 IEEE Security and Privacy Workshops, San Jose, CA, USA, 21–22 May 2015.

17. Fatokun, T.; Nag, A.; Sharma, S. Towards a Blockchain Assisted Patient Owned System for Electronic Health Records. *Electronics* **2021**, *10*, 580. [CrossRef]
18. Jabarulla, M.Y.; Lee, H.-N. Blockchain-Based Distributed Patient-Centric Image Management System. *Appl. Sci.* **2021**, *11*, 196. [CrossRef]
19. Wehbe, Y.; Zaabi, M.A.; Svetinovic, D. Blockchain AI Framework for Healthcare Records Management: Constrained Goal Model. In Proceedings of the 2018 26th Telecommunications Forum (TELFOR), Belgrade, Serbia, 20—21 November 2018; pp. 420–425.
20. Kaaniche, N.; Maryline, L. A blockchain-based data usage auditing architecture with enhanced privacy and availability. In Proceedings of the 2017 IEEE 16th International Symposium on Network Computing and Applications (NCA), Cambridge, MA, USA, 30 October–1 November 2017.
21. Maymounkov, P.; David, M. Kademlia: A peer-to-peer information system based on the xor metric. In *International Workshop on Peer-to-Peer Systems*; Springer: Berlin/Heidelberg, Germany, 2002.
22. Passant, A.; Laublet, P.; Breslin, J.G.; Decker, S. A uri is worth a thousand tags: From tagging to linked data with moat. *Int. J. Semant. Web Inf. Syst. (IJSWIS)* **2009**, *5*, 71–94. [CrossRef]
23. Teutsch, J.; Christian, R. A scalable verification solution for blockchains. *arXiv* **2019**, arXiv:1908.04756.
24. Liang, X. Provchain: A blockchain-based data provenance architecture in cloud environment with enhanced privacy and availability. In Proceedings of the 2017 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID), Madrid, Spain, 14–17 May 2017.
25. Rantos, K.; Drosatos, G.; Demertzis, K.; Ilioudis, C.; Papanikolaou, A. Blockchain-based Consents Management for Personal Data Processing in the IoT Ecosystem. *ICETE* **2018**, *2*, 572–577.