Available online at www.ijournalse.org

Emerging Science Journal

(ISSN: 2610-9182)

Vol. 7, No. 2, April, 2023

Secure Distributed Cloud Storage based on the Blockchain Technology and Smart Contracts

Solonas Gousteris [1], Yannis C. Stamatiou [2], Constantinos Halkiopoulos [1*], Hera Antonopoulou [1], Nikos Kostopoulos [1]

[1] *Entrepreneurship and Digital Innovation Laboratory (EDILAB), Department of Management Science and Technology, University of Patras, Greece.*

[2] *Department of Business Administration, University of Patras, Greece and Computer Technology Institute and Press (CTI) – "Diophantus", Patras, Greece.*

Abstract

Objectives: This paper addresses the problem of secure data storage and sharing over cloud storage infrastructures. A secure, distributed cloud storage structure incorporating the blockchain structure is proposed that supports confidentiality, integrity, and availability. *Methods/Analysis:* The proposed structure combines two well-known technologies: one of them is the Ethereum Blockchain and its Smart Contracts and the other is the RSA encryption and authentication scheme. The Ethereum Blockchain is used as a data structure, which ensures data availability and integrity while RSA provides sensitive data confidentiality and source authentication. *Findings:* As a result, users of the proposed structure can trust it and be certain that they can securely exchange information through a publicly accessible and shared cloud storage. The application can be used either through a user interface (UI) or a command-line interface (CLI). *Novelty /Improvement:* The novelty of this work is that the system that is proposed could be used for secure data storage on the cloud as well as for file sharing and authentication verification. Also, secure data storage and file sharing are already offered by the proposed system.

Keywords:
Ethereum; Blockchain;
RSA; Encryption;
Solidity;
Cloud Storage.

Article History:

Received:	14	July	2022
Revised:	23	November	2022
Accepted:	16	December	2022
Available online:	14	February	2023

1- Introduction

As a result of the wide availability of storage facilities with huge capacities (e.g., data centers) as well as cloud infrastructures over these facilities, distributed information structures and systems (see, e.g., [1]) are commonly used by organizations and businesses for data storage and sharing. Multiple computers and storage resources are connected transparently as a uniformly accessible "single" resource that users can deploy in order to store and share their files and exchange information.

The Blockchain [2], in particular, is one of the most popular openly distributed structures that can be used to securely store, and share information related to cryptocurrency creation and spending [3]. The blockchain structure is composed of blocks forming a linked list. Each block stores transaction data, ensuring the integrity and availability of stored information. The computation nodes are the participants in the blockchain network, and each of them "mines" the cryptocurrency transactions. Specifically, a transaction is «verified» by the node, which was the first one that solved a, computationally, difficult mathematic problem which is computation intensive and requires powerful hardware resources. Consequently, data in a block cannot be modified and cannot be deleted since this hard work must be repeated for numerous blocks in the blockchain, which is computationally infeasible. Thus, the blockchain, by design, ensures the integrity and continuous availability of the stored data.

* **CONTACT**: halkion@upatras.gr

DOI: http://dx.doi.org/10.28991/ESJ-2023-07-02-012

The blockchain's first application domain was digital currency, in particular *bitcoin*, which is a decentralized cryptocurrency based on the peer-to-peer, distributed, and integrity properties of the blockchain structure. Due to these properties of the blockchain, cryptocurrency does not require a centralized regulatory body. The monetary transactions are stored in the blockchain structure, and they are secured by the participating peer network nodes through cryptographic constructs. New bitcoins are constantly being issued and given as a reward to those who confirm transactions (called "miners").

Over the years, however, it has been realized that blockchain has application potential for several application domains beyond cryptocurrency. It is, indeed, a general decentralized, peer-to-peer based, mechanism where existing information is verifiable through complex cryptographic units of computational work. Although the market for digital cryptocurrency led to the invention and development of blockchain technology, the blockchain structure can store, virtually, any kind of information that requires a high level of integrity and verifiability. Specifically, with respect to storage of persistent and immutable information, the blockchain is one of the best available distributed solutions. However, it cannot ensure the confidentiality of stored data since blockchain information is open and publicly available. Therefore, a user cannot trust the blockchain structure to share sensitive information with other users. Therefore, a secure method (see [4]) should be introduced in the blockchain structure to ensure that users' sensitive data can be shared with only the intended recipients in an integrity and confidentiality preserving manner. The well-known RSA public key encryption algorithm (see, e.g., [5]) can be used (among others), both for authentication and data confidentiality. It is an asymmetric encryption method that uses the public/private key pair approach, where data can be encrypted using the public key of a user and be decrypted by the public-key's pair, the private key. The public key (unlike the private) can be visible to everyone in a network.

Our work proposes a combination of the confidentiality offered by the RSA public key encryption scheme with the integrity and sharing, peer-to-peer features of the blockchain structure. The proposed scheme was also implemented, producing a shared cloud storage construct that can be deployed for secure, integrity-preserving, and source-verifiable data sharing over the Internet as a cloud service.

In contrast with all the recent related works we review in Section 2.1, our proposal has the following novel elements:

- It deploys a strong public key encryption algorithm, RSA, on a 2048-bit key, that provides strong data confidentiality as well as data authentication through the public keys that are verifiably uploaded in Ethereum smart contracts. RSA works as a plug-in in the proposed scheme, and, thus, it is easily replaceable by some other public key algorithm if necessary.
- Files of unlimited size can be uploaded since they are divided into several encrypted, but chained in the correct order, pieces distributed over the Ethereum blocks – Ethereum operation also provides a file integrity functionality.
- Files can be shared with any user as long as he/she uploads his/her public key into a smart contract and performs the file sharing protocol we define.

2- Literature Review and Our Proposal

2-1- Background and Literature Review

The blockchain falls into the generic system category of *distributed systems*. A distributed system is a collection of standalone computers connected by a network infrastructure in order to deliver, collectively, integrated computational services. These computers may operate independently of one another. These independent computers, which are often referred to as *nodes*, can be either devices (as in the case of the *Internet of Things*) or *processes* (computational tasks that run automatically or manually). The nodes exchange "health check" messages, also known as *health checks*, to ensure the correct operation of the distributed system. In any case, the distributed system's objectives remain unambiguous and unaltered. These objectives include *adaptability*, *performance*, *reliability*, and *accessibility*.

The blockchain as a distributed, shared structure, supports, essentially, a series of peer-to-peer, distributed transactions (e.g., purchases from e-shops). Each new group of entries, called a *block*, is chained to the entries of the previous block, creating a "chain" of grouped transaction entries, which is the "blockchain". The blocks arise through a process in which the algorithmic solution to a computationally hard problem is computed, as described below. In this way, the blockchain functions as a *peer-to-peer distribution system* known to all users, ensuring the security and transparency of transactions. Thus, it is no longer necessary to have an intermediate "trust" authority (e.g., a bank), while trust in the participants is based on algorithmic and indisputable confirmations.

Based on the blockchain, for cryptocurrency and monetary transactions, in particular, each paying user transfers one or more cryptocurrency units to the receiver as follows: the payee signs the hash of the previous transaction in the chain along with the recipient's public key and adds the result to the end of the current block, enabling confirmation by the recipient. *Double spending* is prevented as follows: transactions are made public to everyone, and there is a system in which participants agree on a version of the order in which each transaction has been executed (the *longest* blockchain rule). The receiver requires proof, at the time of the transaction, that the majority of the blockchain nodes accept that it was indeed the first transaction, preventing, thus, any attempt at double spending.

This approach requires a value that, when entered as input to a *hash function*, causes its output to start with a certain number of leading zeros. Therefore, the work (i.e., the computational "effort") required to find this hash value is exponential in the number of zeros. The more zeros required; the more exponentially costly work is required to obtain the required hash input. The proof of this work is provided by increasing a value inside the block until a value is found that leads to a hash value having the required number of leading zeros. Once the computational effort has been used to satisfy the proof of work, the block cannot be changed without expending a formidable amount of computational effort to, essentially, recompute the hash values of the *whole* blockchain. Thus, when a transaction block is "buried" sufficiently deeply in the chain, it can be considered confirmed, or "locked".

Since the transaction (block) information is distributed in multiple copies over a large network of peers (computing notes), it is very difficult for external, *malicious* entities to interfere with the operation of the blockchain since there is no *single point of failure*. The nodes in the network are also not owned by any single institution or organization (e.g., banks, large companies, political parties, or countries, etc.), thus ensuring a high level of transparency. This makes the blockchain a suitable technology for storing information that requires a high level of integrity and verifiability (as in our application, which will be discussed later).

Public visibility and access to the blockchain ensure the availability and transparency of transactions as well as the dissemination of information. In the same context, the audit process is facilitated by eliminating any possible infringements precisely because of the public nature of the data. At the same time, the need for intermediary parties that increase costs and decrease trust disappears since all information related to the transaction is encrypted within the blockchain. However, each participant can see the transactions without further details. Therefore, data stored in the blockchain has the properties of availability and integrity by default. The term "*data availability"* means that the data is always available, while the term integrity means that the data remains immutable under any circumstances and at any time. Although the blockchain is considered reliable distributed storage, some vulnerabilities and potential attacks have been reported (see [6, 7]).

With respect to work related to ours, He et al. [8] propose an Access Control Scheme for sharing the data on the cloud using an attribute-based hierarchical data access control scheme. However, the use of blockchain technology is referred to, in the paper, as future work. Then, Wang et al. [9] propose an Access Control Scheme for sharing data based on ciphertext-policy attribute-based encryption (CP-ABE) on the Ethereum platform. However, this scheme allows for sharing of the encryption key among several users while encrypted data is stored in the Ethereum contract structure, which precludes its use for other purposes. Wang et al. [10] propose a privacy-guaranteed and user-controlled data exchange scheme for IoT devices using fine grained and attribute-based encryption (CP-ABE) in Ethereum Smart Contracts. Their scheme is particularly tailored for IoT devices, whose processing power limitations preclude their use as distributed data storage, much like "multi-element" cloud structures. Qin et al. [11] propose an access control scheme for secure data sharing on the cloud based on a multi-authority approach and Shamir's secret sharing scheme on the Hyperledger Fabric platform[*]. This proposal, however, suffers from the secret sharing requirement, which is beset by the key management issues inherent in key sharing schemes.

Zuo et al. [12] propose a secure data sharing scheme on the cloud, without any third trusted parties, based on ciphertext-policy attribute-based encryption (CP-ABE) and blockchain technology. Zhu et al. [13] propose a management scheme for digital assets using transaction-based access control and attribute-based encryption (CP-ABE), on the blockchain. This scheme focuses mostly on asset access permission management rather than encrypted data sharing. Di Francesco Maesa et al. [14] propose an auditable access control system based on a generic access control scheme on the Ethereum platform. However, in this proposal, the authors focus on transaction auditability due to the immutability and transparency properties of the Ethereum blockchain structure and not on encrypted data sharing. Gao et al. [15] propose a trustworthy, secure data exchange scheme for policy based on blockchain-based access control. In this proposal, the blockchain structure is deployed for traceability and accountability purposes since users can prove the authority of data using smart contracts. Again, in this work, encrypted data sharing is not the focus. Sun et al. [16] deploy a ciphertext-based encryption scheme for electronic medical records access based on the decentralized InterPlanetary File System (IPFS) and blockchain technology. However, this proposal deploys a separate system (the IPFS) beyond the blockchain and, thus, becomes technology dependent. The same disadvantage exists in the proposals of Zhang et al. [17] and Steichen et al. [18] for a decentralized access control mechanism for data sharing on the Ethereum platform. Arthur Sandor et al. [19] propose a decentralized system for mobile cloud data storage based on a multi-authority access scheme. The focus here is not encrypted file exchange. Stanciu [20] proposed an Edge Computing control system based on the Hyperledger Fabric platform. Again, the focus is not on encrypted files and data sharing. Finally, blockchain multiparty access control schemes, but with no emphasis on encrypted file sharing, are proposed by Zhu et al. [21], Paillisse et al. [22], and Guo et al. [23].

In the next subsection we provide the key elements of our approach.

[*] *https://www.hyperledger.org/use/fabric https://www.hyperledger.org/use/fabric*

2-2- Our proposal: The Ethereum Blockchain and the RSA Algorithm: Our Main Building Blocks

Ethereum is an open blockchain platform that enables the creation, execution, and use of decentralized applications and smart contracts using the Solidity programming language. Typical is the facilitation and automation it offers to the interaction between the participants of the network. Ether (ETH) is the currency of the implemented application and works as a price for mining in the Ethereum network.

Ethereum was released in July 2015. It is a decentralized platform that uses the so-called "smart contracts" concept. Smart contracts are self-activating contracts or applications that work exactly as planned without any possibility of not executing (i.e., the Ethereum blockchain is never down and always running), censorship, fraud, or third-party interference. Ethereum has a capability that goes far beyond the properties of a pure P2P digital currency such as Bitcoin. Ethereum is very similar to a smartphone operating system on which software applications can be created (see [8]).

From a technical point of view, the Ethereum platform itself is not a cryptocurrency. However, like other open, unauthorized blockchains, Ethereum requires a value chain within the chain to enable on-net transaction validation (i.e., a payment method for network nodes performing the operations). This is where Ethereum (ETH) cryptocurrency comes into play. Ether not only allows you to build smart contracts on the Ethereum platform (i.e., feed them), but also acts as a medium of exchange. Like Bitcoin, Ethereum currently uses a PoW (Proof of Work) consent mechanism but is slowly moving towards a PoS (Proof of Stake) consent mechanism, known as the Casper protocol. The development of Ethereum is promoted and supported by the Ethereum Foundation, a Swiss non-profit organization founded by the inventors of Ethereum. A large volume of Ethereum was "pre-mined" (i.e., mined / created before the official release of the currency to the public) by its inventors and sold in large numbers to pay for development costs and to fund the Ethereum Foundation. Just like Bitcoin, Ethereum is a prime example of an open, unauthorized Blockchain. Anyone can join or leave the Ethereum network at will without having to be pre-approved by any central entity, and like Bitcoin, Ethereum (ETH) can be categorized as a pseudo-anonymous currency).

On the other hand, with respect to data confidentiality, the RSA (Rivest-Shamir-Adleman) crypto algorithm is one of the first public-key encryption algorithms and is widely used for secure data transmission. It is based on the difficulty of factoring large numbers (today, usually in the range of 1024 to 2048 bits). Two keys are used, one public during encryption and one private for decryption. The acronym RSA was derived from Ron Rivest, Adi Shamir, and Leonard Adleman, from whom the algorithm was first published in 1977.

One of the most basic (perhaps the most basic) applications that the algorithm encounters is encrypting data sent by browsers. The well-known HTTPS protocol differs from that of HTTP in that the former encrypts the data before the server accepts it. The network layer where this encryption takes place is called the transfer layer, and it takes over this process. Thus, for each web page that is accessed through the HTTPS protocol, for example when a user registration form is filled in, the data is encrypted before it reaches the server so that the "channel" that connects them is secure and no one can access information from a malicious user. The attack, as mentioned earlier, is called a man-in-the-middle attack (Figure 1) and describes the process by which someone managed to appear between a client-server and read any information they exchanged.

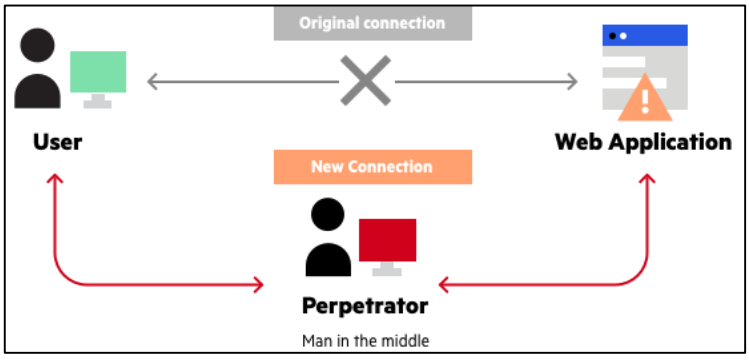

Figure 1. The Man-in-the-middle attack

Nevertheless, the RSA algorithm still solves the problem through the reliable encryption process it provides. It is worth noting, of course, that for the above use of RSA, not only a pair of keys is used but also some other entities (e.g., the Certificate Authority), which make communication even more reliable. In short, they certify that the keys were generated by a reliable source and for the specific website (URL) that is accessed.

3- The Proposed System

3-1- The Blockchain Perspective

It is well known that the blockchain structure can be exploited in several application domains [24, 25, 26]. The most popular one is the cryptocurrency exchange, where multiple users can interact with the system. This paper proposes a

solution for the problem by taking advantage of blockchain as a data structure, where data is available and remains unmodified by default.

In general, our belief (as it will become apparent from the application we describe in this paper) is that the blockchain can be organized (for the application type we describe) into something like an ADT (Abstract Data Type) in the context data structures that store information. An ADT is the description of a data storage structure plus the operations allowed on it, much like a stack or a tree data structure, for instance. We may have an abstraction of the blockchain with a corresponding generic, high level, API that will support high-level operations such as "Add record", "Verify transaction", "Search for transaction", "find the longest transaction chain between two owners", "find the shortest path between two transactions," etc. In this way, several types of applications [27, 28] can deploy the blockchain in order to have a seamless integration layer with the application level (i.e., a transparent application/security level interaction) that will form a layer on which all application domains will be based for building trust, privacy, and security for their users.

3-2- The System Architecture

The proposed system (Figure 2) uses the Ethereum Blockchain [29] and the Smart Contracts [30] for user-system interaction.

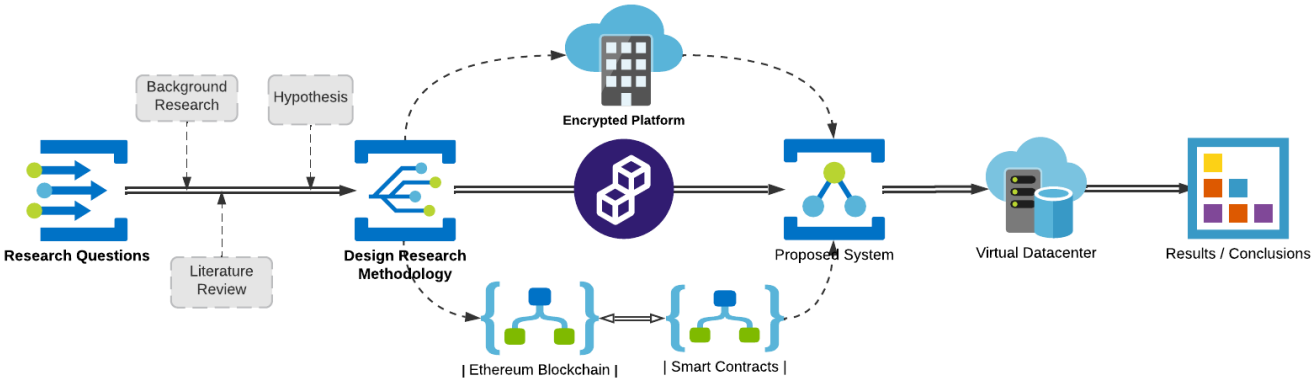

Figure 2. Flowchart of Proposed Methodology/System

The system's operation is divided into four tasks:

- **1st Task - RSA key generation:**

  The user must, initially, generate his/her public-private RSA key pair and upload the public key, along with a unique username, to the blockchain. The following figure (Figure 3) describes the process for creating and storing RSA keys. Two ".pem" type files are generated, one of which containing the private key ("private.pem") and the other the public key ("public.pem"). Then, the user selects the folder destination in his/her local machine in which these files will be stored. During this first step, the user does not interact, yet, with the blockchain.

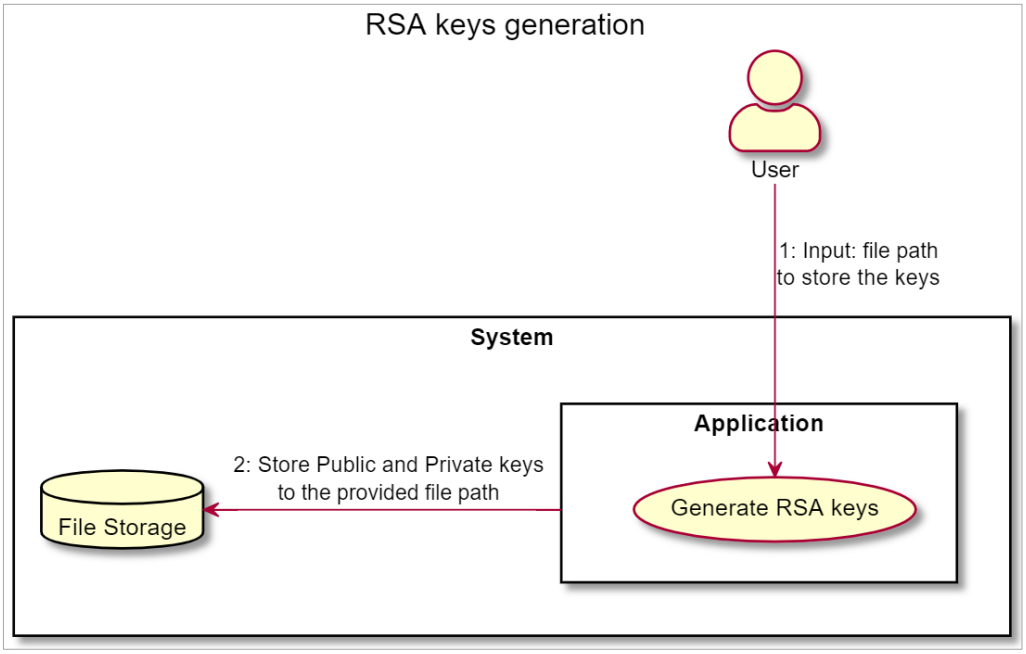

Figure 3. RSA Key Generation (2048 bits)

- **2nd Task - User registration**

Subsequently, the user can use the smart contract functionality to upload, in a certified way, his/her personal information on the blockchain, which will be public and available to all users. This contract will be used by anyone who wants to send files to this user. This presupposes the user registration process shown in Figure 4.

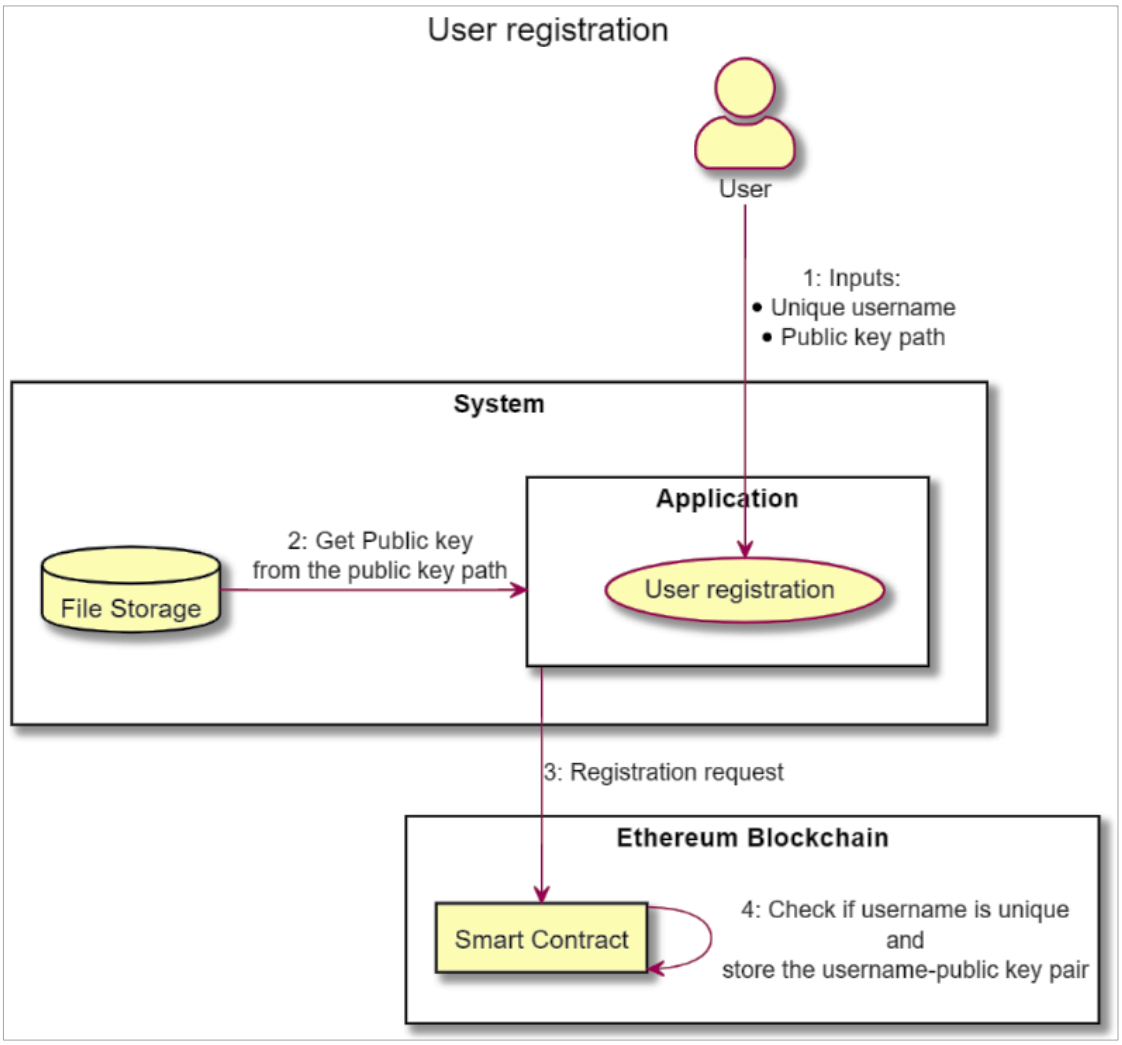

Figure 4. Flowchart of "User Registration"

Each user enters, in the contract, the following as input:

- A username of his/he choice
- The public key generated in the previous step (i.e., the file "public.pem" produced by existing RSA key generation tools* or other certified methods if the stored information is critical)

The process is performed entirely within the blockchain, from the point of view that even the control is done during the execution of the smart contract (solidity). The same username cannot be entered twice, which is why this check is performed (Figure 4).

- **3rd Task - File uploading**

The process of file uploading, shown in Figure 5, and file sharing are implemented as follows. The user deploys this functionality to share a file with the desired username in a secure way using RSA encryption. Suppose that User "A" wants to share a file with User "B". User "A" selects the desired file. Then, using the smart contract operations implemented, the system asks the public key of User "B" to encrypt the file and ensure that only User "B" can decrypt it using his private key securely stored on his local machine.

*See, for instance, https://gist.github.com/fictorial/184283/fc18c2d3e5cfba7fd5bfa80b870e23f5e66fa9c0

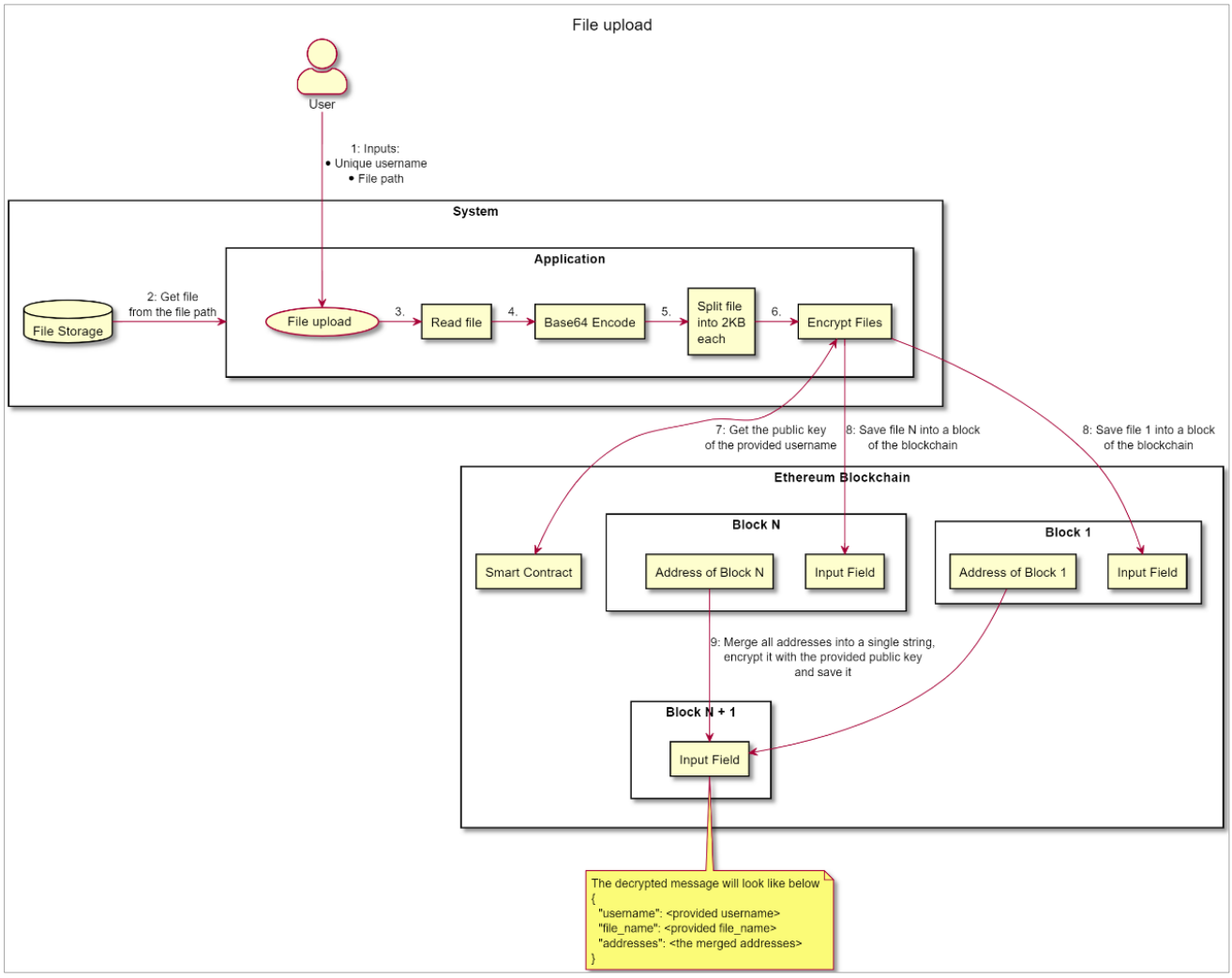

Figure 5. Flowchart of "Uploading a file"

The steps implementing this functionality follow below:

First step

The user enters as input:

1. The username of the user to whom he/she wants to send the file, i.e., the username of the recipient;
2. The destination of the file it wishes to share/send.

Second Step

The program reads the file and encodes it using the Base64 representation. This encoding makes it possible to send any file of any conceivable format, such as ".txt", ".png" or ".pdf" for instance.

Third Step

The file is split into smaller parts because it is stored in the input field of a block in the Ethereum blockchain and the space available in this field is limited. Each piece of the file is then encrypted with RSA using the recipient's public key, which has been made known through a functionality of Smart Contracts. Finally, the address of the block in which each piece of the file was stored is returned.

Fourth Step

After collecting all the addresses of the blocks in which the split pieces were stored, the application creates an object with JSON (JavaScript Object Notation) format which contains the username, file name, and all the addresses of the blocks which store the file pieces. Finally, it encrypts this object with the same key and saves it in a new block.

In summary, in the end the Ethereum blockchain will contain a block which stores the following information (in the encrypted JSON object we discussed before):

- The recipient's username;
- The file's name;
- The addresses (in the correct order) of the blocks containing the encrypted data for each piece of the original file transmitted by the user.

Additionally, all these information items are encrypted so that only the authorized recipient can find the locations (blocks) of the file pieces. The authorized user is the one who possesses the private key that corresponds to his/her public key.

- **4th Task - File downloading**

Let us now describe the file-downloading process scenario (Figure 6). User "B" triggers the downloading process through the smart contract and selects the file that he/she wants to download. The system provides the selected file in encrypted form, and only after the download has finished does the decryption process start automatically using the provided private key.

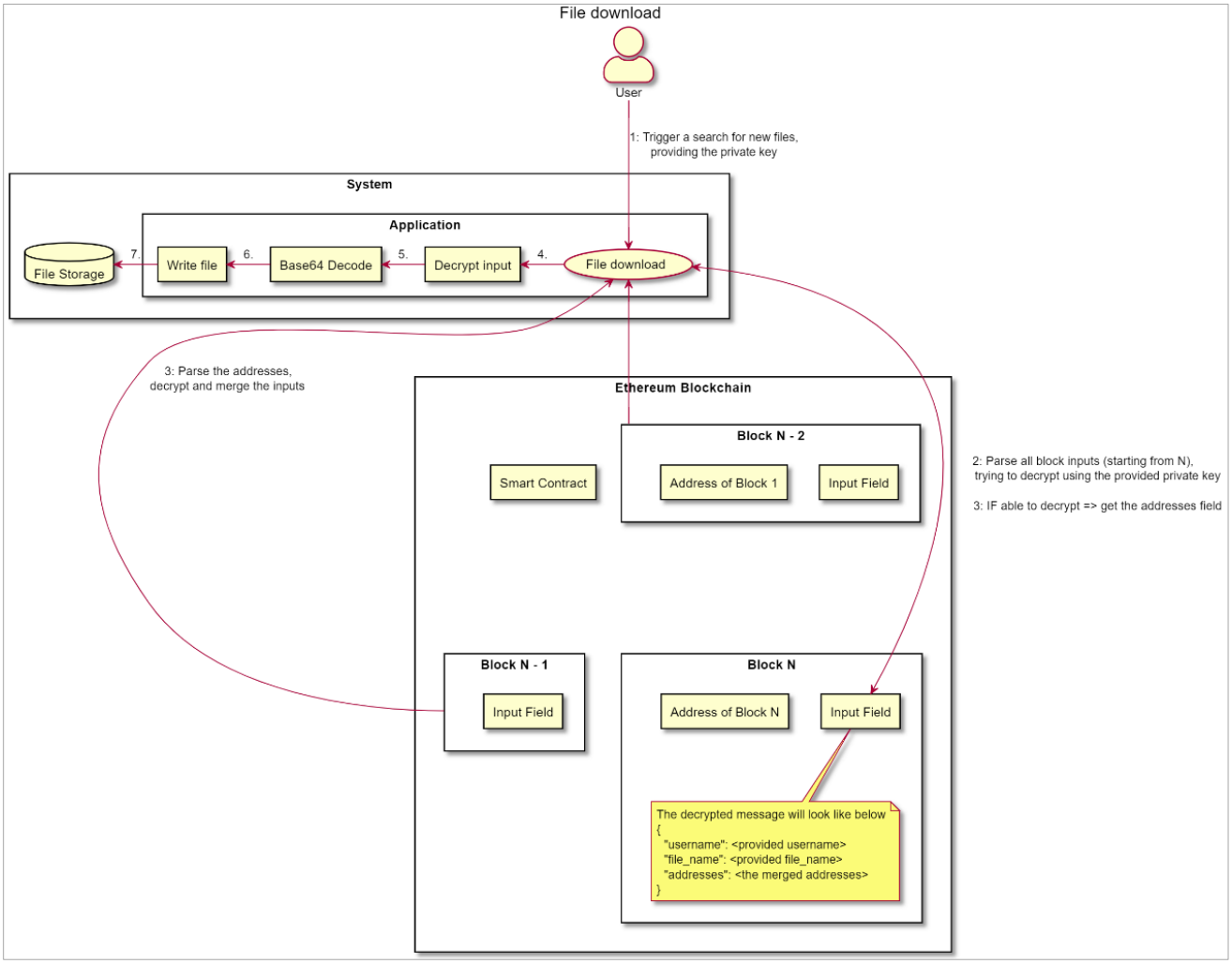

Figure 6. Flowchart of "Downloading a File"

In some more detail, initially the user enters his username and his private key to read only the blocks that he/she is allowed to access and decrypt. The private key is also used as an authentication token for the user since only this key can decrypt information encrypted in the corresponding blocks with his/her public key.

By decrypting the block containing the addresses of the file pieces (i.e., the blocks that store them), the user obtains, from the "Addresses" field, all file pieces in the correct order.

Next, the process accesses and decrypts the blocks one by one, obtaining, in this way, all the file pieces that are spliced together in order to produce the sent file. The file pieces are encoded using the base64 encoding, and, thus, the last step is to convert from base64 to binary to obtain the original file.

3-3- Performance Estimation

In general, since the proposed system relies on the blockchain structure, it inherits its performance characteristics. The downloading feature, in particular, is intended to search the blockchain structure in response to an information request. Thus, the search operation requires time proportional to the overall structure's size. More specifically, the search algorithm starts from the most recent block and moves towards the first block in the blockchain structure, and, thus, searching can be accomplished in several block access operations equal, in the worst case, to the number of blocks in the blockchain. However, searching in the blockchain can be, easily, parallelized since one can engage several nodes in possession of the current blockchain structure and assign to them disjoint, contiguous, sub-chains within which they can search, in parallel, for the requested information.

In addition, data encryption (for the uploaded file pieces) based on RSA is, rather, slow and becomes impracticable for very long files. More specifically, below there are listed some examples of a pilot testing on a Docker container:

- Key creation: 00:00,68;
- Registering: 00:04,96;
- File Encrypting & Uploading 4.0K file: 00:08,80;
- File Decrypting & Downloading (across 32 blocks:
  - Parsing Blockchain to check available downloads: 00:02,26;
  - Actual Downloading 00:00,60;

Let’s see how the above results are influenced by the Blockchain length. Thus, we’ll leave a mining process to take place for 1-2 minutes:

The new Blockchain length is 97 (~3 times more than the previous testing).

- Key creation: 00:00,66;
- Registering: 00:05,68;
- File Encrypting & Uploading 4.0K file: 00:07,85;
- File Decrypting & Downloading (across 32 blocks) using same user as the previous step:
  - Parsing Blockchain to check available downloads: 00:03,03;
  - Actual Downloading 00:00,81.

The "Parsing Blockchain" step is the one that is most influenced by the Blockchain length. That’s expected, since parsing works by querying Blockchain from the latest to the first block. Thus, if a user’s data was stored on an old block and the chain’s length is big, then this step would take much more time compared to other steps.

3-4- Pilot Testing

The system was tested on a Virtual Machine based on an Ubuntu 20.04 image. The required tools had been installed, and the blockchain server was deployed on the Virtual Machine, configuring the difficulty value of generating a new block to a relatively low value for testing purposes. Both the UI and CLI applications were tested with a limited amount of data and two users: one acting as the sender and the other as the receiver of the shared secret. The secret was a simple ".txt" file containing a database access key with root privileges. Since the amount of data was limited, the whole process (including uploading, downloading, as well as encryption/decryption) was completed within approximately 3 minutes.

4- Conclusion

The security of stored information is a major issue in cloud and data sharing infrastructures. The system we propose as another application domain of the blockchain (Ethereum) can be used as a secure peer-to-peer data storage on the cloud as well as for file sharing and file authenticity verification. Also, secure data storage and file sharing are offered by the proposed system. Our viewpoint is that we need to address potential alternative uses of distributed data structures offering many security functionalities in a peer-to-peer context, such as the blockchain, beyond monetary or other types of commodity transactions (e.g., smart contracts).

As further work, we plan to implement the *authenticity verification* functionality (the digital signature approach based on RSA), where the use case is the following: assuming that the User "A" wants to obtain a strong belief that an uploaded document belongs to the real User "B" that uploaded it. In that case, the User "B" should upload the document encrypted with his own private key and let all users verify the document’s authenticity and decrypt it with the User "B" public key, which is available to anyone. Further work on other similar functionalities is also among the plans of our team for subsequent work.

5- Declarations

5-1- Author Contributions

Conceptualization, I.S., S.G., C.H., H.A., and N.K.; methodology, I.S., S.G., C.H., H.A., and N.K.; writing—original draft preparation, I.S., S.G., C.H., H.A., and N.K.; writing—review and editing, I.S., S.G., C.H., H.A., and N.K. All authors have read and agreed to the published version of the manuscript.

5-2- Data Availability Statement

The data presented in this study are available on request from the corresponding author.

5-3- Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

5-4- Institutional Review Board Statement

Not applicable.

5-5- Informed Consent Statement

Participants gave their written consent to use their anonymous data for statistical purposes. All of them were over 18 years old and voluntarily collaborated without receiving any financial compensation.

5-6- Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this manuscript. In addition, the ethical issues, including plagiarism, informed consent, misconduct, data fabrication and/or falsification, double publication and/or submission, and redundancies have been completely observed by the authors.

6- References

[1] Coulouris, G. F., Dollimore, J., & Kindberg, T. (2005). Distributed systems: concepts and design. Pearson Education, New York City, United States.

[2] Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. Decentralized Business Review, 21260. Available online: https://www.debr.io/article/21260-bitcoin-a-peer-to-peer-electronic-cash-system (accessed on January 2023).

[3] Lee, D. K. C., Guo, L., & Wang, Y. (2018). Cryptocurrency: A new investment opportunity? Journal of Alternative Investments, 20(3), 16–40. doi:10.3905/jai.2018.20.3.016.

[4] Yu, S. (2010). Data sharing on untrusted storage with attribute-based encryption. Ph.D. Thesis, Worcester Polytechnic Institute, Worcester, United States.

[5] Rivest, R. L., Shamir, A., & Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM, 21(2), 120–126. doi:10.1145/359340.359342.

[6] Conti, M., Sandeep Kumar, E., Lal, C., & Ruj, S. (2018). A Survey on Security and Privacy Issues of Bitcoin. IEEE Communications Surveys & Tutorials, 20(4), 3416–3452. doi:10.1109/comst.2018.2842460.

[7] Titcomb, j. (2017). US regulator warns of 'fraud and manipulation' amid cryptocurrency craze. Daily Telegraph. Available online: http://www.telegraph.co.uk/technology/2017/12/12/us-regulator-warns-fraud-manipulation-amid-cryptocurrency-craze/ (accessed on January 2023).

[8] He, H., Zheng, L., Li, P., Deng, L., Huang, L., & Chen, X. (2020). An efficient attribute-based hierarchical data access control scheme in cloud computing. Human-Centric Computing and Information Sciences, 10(1), 1-19. doi:10.1186/s13673-020-00255-5.

[9] Wang, S., Wang, X., & Zhang, Y. (2019). A Secure Cloud Storage Framework with Access Control Based on Blockchain. IEEE Access, 7, 112713–112725. doi:10.1109/ACCESS.2019.2929205.

[10] Wang, S., Zhang, Y., & Zhang, Y. (2018). A blockchain-based framework for data sharing with fine-grained access control in decentralized storage systems. IEEE Access, 6, 38437–38450. doi:10.1109/ACCESS.2018.2851611.

[11] Qin, X., Huang, Y., Yang, Z., & Li, X. (2021). A Blockchain-based access control scheme with multiple attribute authorities for secure cloud data sharing. Journal of Systems Architecture, 112, 101854. doi:10.1016/j.sysarc.2020.101854.

[12] Zuo, Y., Kang, Z., Xu, J., & Chen, Z. (2021). BCAS: A blockchain-based ciphertext-policy attribute-based encryption scheme for cloud data security sharing. International Journal of Distributed Sensor Networks, 17(3), 1550147721999616. doi:10.1177/1550147721999616.

[13] Zhu, Y., Qin, Y., Zhou, Z., Song, X., Liu, G., & Chu, W. C.-C. (2018). Digital Asset Management with Distributed Permission over Blockchain and Attribute-Based Access Control. 2018 IEEE International Conference on Services Computing (SCC). doi:10.1109/scc.2018.00032.

[14] Di Francesco Maesa, D., Mori, P., & Ricci, L. (2019). A blockchain based approach for the definition of auditable Access Control systems. Computers & Security, 84, 93–119. doi:10.1016/j.cose.2019.03.016.

[15] Gao, S., Piao, G., Zhu, J., Ma, X., & Ma, J. (2020). TrustAccess: A Trustworthy Secure Ciphertext-Policy and Attribute Hiding Access Control Scheme Based on Blockchain. IEEE Transactions on Vehicular Technology, 69(6), 5784–5798. doi:10.1109/TVT.2020.2967099.

[16] Sun, J., Yao, X., Wang, S., & Wu, Y. (2020). Blockchain-Based Secure Storage and Access Scheme for Electronic Medical Records in IPFS. IEEE Access, 8, 59389–59401. doi:10.1109/ACCESS.2020.2982964.

[17] Zhang, Y., He, D., & Choo, K. K. R. (2018). BaDS: Blockchain-based architecture for data sharing with ABS and CP-ABE in IoT. Wireless Communications and Mobile Computing, 2018. doi:10.1155/2018/2783658.

[18] Steichen, M., Fiz, B., Norvill, R., Shbair, W., & State, R. (2018). Blockchain-Based, Decentralized Access Control for IPFS. 2018 IEEE International Conference on Internet of Things (IThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData). doi:10.1109/cybermatics_2018.2018.00253.

[19] Arthur Sandor, V. K., Lin, Y., Li, X., Lin, F., & Zhang, S. (2019). Efficient decentralized multi-authority attribute-based encryption for mobile cloud data storage. Journal of Network and Computer Applications, 129, 25–36. doi:10.1016/j.jnca.2019.01.003.

[20] Stanciu, A. (2017). Blockchain Based Distributed Control System for Edge Computing. 21st International Conference on Control Systems and Computer Science (CSCS), Bucharest, Romania. doi:10.1109/cscs.2017.102.

[21] Zhu, L., Wu, Y., Gai, K., & Choo, K. K. R. (2019). Controllable and trustworthy blockchain-based cloud data management. Future Generation Computer Systems, 91, 527–535. doi:10.1016/j.future.2018.09.019.

[22] Paillisse, J., Subira, J., Lopez, A., Rodriguez-Natal, A., Ermagan, V., Maino, F., & Cabellos, A. (2019). Distributed Access Control with Blockchain. ICC - 2019 IEEE International Conference on Communications (ICC). doi:10.1109/icc.2019.8761995.

[23] Guo, H., Meamari, E., & Shen, C.-C. (2019). Multi-Authority Attribute-Based Access Control with Smart Contract. Proceedings of the 2019 International Conference on Blockchain Technology. doi:10.1145/3320154.3320164.

[24] Antonopoulou, H., Giannoulis, A., Theodorakopoulos, L., & Halkiopoulos, C. (2022). Socio-Cognitive Awareness of Inmates through an Encrypted Innovative Educational Platform. International Journal of Learning, Teaching and Educational Research, 21(9), 52–75. https://doi.org/10.26803/ijlter.21.9.4

[25] Stamatiou, Y. C., Halkiopoulos, C., Giannoulis, A., & Antonopoulou, H. (2022). Utilizing a Restricted Access e-Learning Platform for Reform, Equity, and Self-development in Correctional Facilities. Emerging Science Journal, 6, 241–252. https://doi.org/10.28991/esj-2022-sied-017

[26] Antonopoulou, H., Halkiopoulos, C., Gkintoni, E., & Katsimpelis, A. (2022). Application of Gamification Tools for Identification of Neurocognitive and Social Function in Distance Learning Education. International Journal of Learning, Teaching and Educational Research, 21(5), 367–400. https://doi.org/10.26803/ijlter.21.5.19

[27] Halkiopoulos, C., Antonopoulou, H., Kostopoulos, N. (2022). Integration of Blockchain Technology in Tourism Industry: Opportunities and Challenges. 9th International Conference of the International Association of Cultural and Digital Tourism (IACuDiT): "Tourism, Travel and Hospitality in a Smart and Sustainable World", 1-3 September, Syros, Greece

[28] Kostopoulos, N., Antonopoulou, H., & Halkiopoulos, C. (2022). Blockchain Technology as an Asset for Innovative Educational Applications: A Systematic Review. EDULEARN22 Proceedings. https://doi.org/10.21125/edulearn.2022.1358

[29] Wood, G. (2014). Ethereum: A Secure Decentralised Generalised Transaction Ledger. Available online: http://gavwood.com/paper.pdf (accessed on January 2023).

[30] Hyperledger (2022). A Blockchain Platform for the Enterprise: Smart Contracts. Available online: https://hyperledger-fabric.readthedocs.io/en/latest/whatis.html?highlight=smart%20contracts#smart-contracts (accessed on January 2023).