Blockchain-Based Long-Term Multisignature

Aprianti Nanda Sari and Trisna Gelar

Department of Computer and Informatics Engineering
Politeknik Negeri Bandung, POLBAN
Bandung, Indonesia
aprianti.nanda@polban.ac.id, trisna.gelar@polban.ac.id

Abstract. The digital signature is well recognized as a prominent use of asymmetric-key cryptography, effectively addressing the limitations associated with handwritten signatures through the preservation of data integrity. The digital signature is characterized by the involvement of two distinct parties. In practical contexts, it is not uncommon for many parties to affix their signatures to a single document. To mitigate the risk of fraudulent individuals submitting signed papers under false identities, the implementation of digital certificates and certificate authorities (CAs) has been introduced. The previous approach introduces a novel challenge arising from the limited duration of digital signature validation. In contrast, blockchain technology, which was introduced by Satoshi Nakamoto in 2008, has had a significant impact on the field of digital archiving. Therefore, this paper explores an alternative application of blockchain technology for the purpose of preserving previously signed documents, enabling their validation over an extended duration. The test results indicate that the suggested technique has effectively fulfilled the functional requirements of a multisignature system, encompassing data integration, non-repudiation, and traceability.

Keywords: digital signature, multisignature, blockchain, data integrity, non-repudiation, traceability.

1. Introduction

One of the most popular implementations of asymmetric-key cryptography is the digital signature, which overcomes the disadvantage of handwritten signatures by maintaining data integrity. Basically, a digital signature has two main phases i.e. signing and verification. The signing phase is completed at the sender's side. With the use of a hash function like the MD5 algorithm, the hash value of a document or file is extracted. Then, this hash value is encrypted by the private key of the sender to yield a digital signature. Finally, the digital signature and its original document are sent to the receiver. On the receiver's side, the verification phase is conducted by comparing the decrypted digital signature with the hash value of the document. If it is identical, then the authenticity and integrity of the received document are guaranteed.

Notably, the digital signature involves two parties. In the real world, however, there are some instances where multiple parties sign the same document. As an illustration, the supervisors, examining committees, head of department, and so forth must all sign the approval page of the academic thesis document. To overcome this problem, Itakura initiated the multi-signature scheme in 1983[1]. Since then, a lot of multi-signature

M. U. H. Al Rasyid and M. R. Mufid (eds.), *Proceedings of the International Conference on Applied Science and Technology on Engineering Science 2023 (iCAST-ES 2023)*, Advances in Engineering Research 230,
https://doi.org/10.2991/978-94-6463-364-1_18

multi-signature methods based on mathematics problems such as discrete logarithm problems, lattice-based problems, elliptic curves, and so on have been proposed [2]–[8].

Using a digital signature or multi-signature might be a promising way to maintain the authenticity and reliability of a signed document. Nevertheless, untrustworthy parties can still submit signed documents while claiming to be anyone. To prevent this kind of failure, a concept of digital certificates and certificate authorities (CAs) have been established.

Digital certificates are bits of information that link a particular public key to an individual or organization (the certificate subject). The content of the digital certificates is maintained and updated by the CA(s). Updating means renewing the validation period of the digital certificates to refuse the back-dating attacks [9]. This approach brings a new problem due to the short period of digital signature validation.

On the other hand, blockchain, since its invention in 2008 by Satoshi Nakamoto [10], has been influential in the digital archiving sector. Hence, in this article, we explore another implementation of blockchain technology to keep information on historical signed documents so that it can be validated over a long period of time.

2. Related Works

2.1. Digital Signature And Multi-Signatures

Cryptography is a way to secure information by scrambling or mapping the plaintext into another form (encrypt) so that it is not easy to read. To scramble or map the plaintext, a certain number is used as a parameter called *key*. Based on the key used, there are two types of cryptography i.e. symmetric and asymmetric cryptography. The digital signature is among the most well-known applications of asymmetric cryptography. It works by encrypting the hash value of a digital asset by using the sender's private key to yield a digital signature.

When a digital signature is attached to a digital asset, it provides assurance that the asset has not been tampered with and that it originated from the claimed sender. It also ensures that the sender cannot deny sending the asset since the signature can be verified by anyone with access to the sender's public key.

2.2. Blockchain

A blockchain is a list record called block. Each block consists of two major components: the data and the hash value of the previous block, except for the first block, which does not have the second component. The illustration of blockchain is shown in Fig. 1. Note that if someone tries to change the data in a block, the resulting hash value will be different, which means the block might not be trustworthy.

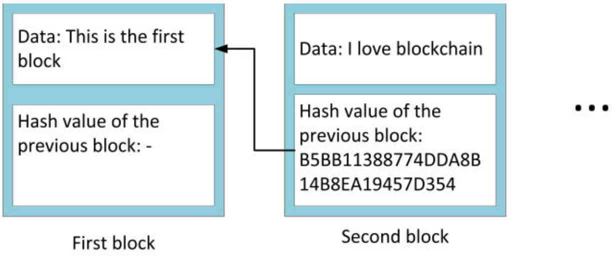

Data: This is the first block
Hash value of the previous block: -
Data: I love blockchain
Hash value of the previous block: B5BB11388774DDA8B 14B8EA19457D354
...
First block
Second block

Fig. 1. Illustration of a blockchain.

In addition, to make this scheme more secure, this blockchain is distributed among the involved participants. Consequently, if an unaccountable entity seeks to substitute a block within the blockchain, it is imperative for this entity to replace the block across all nodes. Therefore, the manipulation of data on the blockchain is exceedingly difficult.

According to Uddin et al., blockchain is comprised of five distinct layers, which are the application layer, data layer, consensus layer, network layer, and execution layer[11]. The specific characteristics and components associated with each of the five layers are shown as follows.

- The application layer defines how users interact with the blockchain system. Possible examples of an application layer include smart contracts, Chaincode, decentralized applications (DApps), and web-based user interfaces.
- The data layer is responsible for establishing the data structure of a block.
- The consensus layer is responsible for specifying the consensus algorithm that is employed. Consensus or agreement algorithms facilitate the collaboration of a cluster of computers in a manner that enables their continued operation in the event of individual member failures [12]. It is important because in a blockchain system, numerous computers are engaged, occasionally including potential eavesdroppers. The consensus algorithm establishes a system aimed at mitigating the potential interference of eavesdroppers in compromising the integrity of the created blockchain. The most used consensus algorithms are Proof-of-Work, Proof-of-Stake, and Proof-of-Authority.
- The network layer is responsible for establishing and managing communication between nodes or computers inside a blockchain system.
- The execution layer assumes the responsibility of performing transactions that have been invoked by the user via the application layer.

3. Research Method

The present study begins by undertaking an analysis of the shortcomings associated with current multi-signature techniques. Subsequently, an extensive review of the academic literature pertaining to blockchain technology is conducted. The present literature investigation has provided valuable insights into the blockchain system, which has served as the foundation for studying the development of the chosen blockchain architecture. Once the blockchain architecture has been established, the subsequent phase involves formulating the methodology for the process of signing and verifying a multi-signature. Finally, the Python programming language was utilized for the execution of the prototype. Finally, the prototype testing is conducted to ensure the prototype meets the functional requirement. Those following steps are illustrated in Fig. 2.

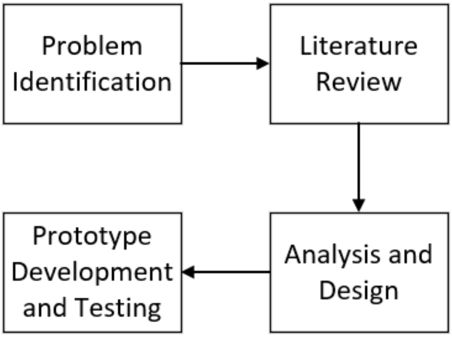


Fig. 2. Research method.

4. Result and Discussion

4.1. Blockchain Architecture

As per the National Institute of Standards and Technology (NIST), a key attribute of a digital signature is its ability to uphold data integrity and non-repudiation [13]. In contrast, multisignature entails the inclusion of traceability as an additional need, in addition to the previous three features [14]. Therefore, Table 1 presents the functional system needs that are to be constructed and their correlation with the properties of blockchain.

Table 1. Functional requirements of the proposed method.

Require-ments	Characteristics of the Blockchain
Data integrity	The blockchain technology possesses characteristics of decentralization and immutability, enabling it to effectively uphold the integrity of data. The reason for this is that every block is linked to the preceding block through the utilization of a hash value. This characteristic is proven by several studies [10], [15]–[17].

Require-ments	**Characteristics of the Blockchain**
Non-repudiation	Once an individual adds a block to the blockchain, their authorization cannot be refuted due to the utilization of private key signatures for each block as shown in several studies [18]–[20].
Traceablity	The utilization of blockchain technology enables a reverse tracing of recorded data, similar to the methodology employed in these research[21]–[24].

Based on those functional requirements as seen in Table 1, the proposed blockchain architecture can be seen in Fig. 3.

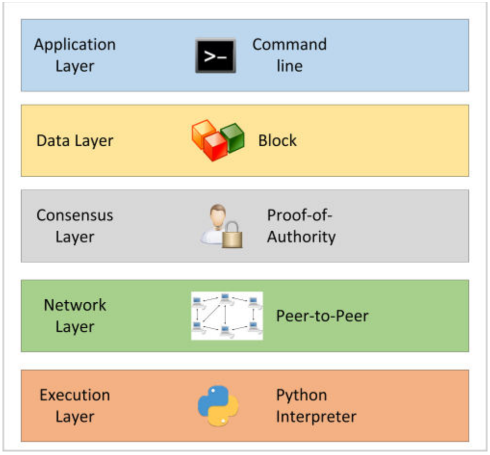


Fig. 3. Blockchain architecture of the proposed method.

The architectural specifics are outlined as follows.

- Application layer, the current iteration of the prototype remains limited to a console or command line interface.
- Data layer (visually shown in Fig. 4), a prototype is represented as a data block includes:
  - *Index*, refers to the numerical value assigned to a specific block.
  - *Message,* contains the hash value of the signed document.
  - *Signer,* refers to the one who possesses the signing username.
  - *Timestamp,* indicates the moment at which the document was signed.
  - *Previous hash* refers to the hash value of the preceding block.

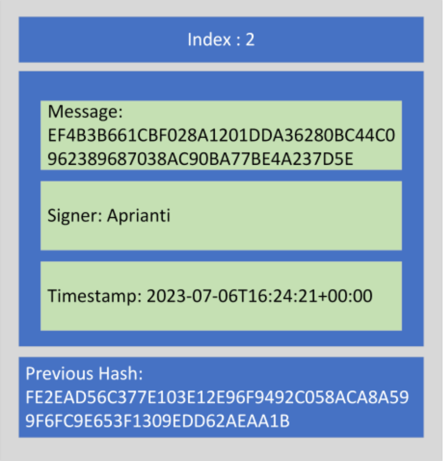

Index : 2
Message:
EF4B3B661CBF028A1201DDA36280BC44C0
962389687038AC90BA77BE4A237D5E
Signer: Aprianti
Timestamp: 2023-07-06T16:24:21+00:00
Previous Hash:
FE2EAD56C377E103E12E96F9492C058ACA8A59
9F6FC9E653F1309EDD62AEAA1B

Fig. 4. Illustration of a block in the proposed data layer.

- Consensus layer, the method used for consensus mechanism is Proof-of-Authority, under the assumption that internal documents within an institution undergo multisignature verification.
- Network layer, the network layer of the prototype operates on a peer-to-peer architecture, enabling all units or unit representations within the institution to have access to the blockchain.
- Execution layer, the implementation of the prototype execution layer is accomplished through the utilization of the Python programming language.

A. *Multisignature Scheme*

The explanations for several notations employed in this paper can be found in Table 2.

Table 2. Notation list.

Notations	**Explanation**
$hash(a)$	The process of obtaining the hash value of document a using the hash function such as MD5 [25], SHA-1 [26], or SHA-3[27].
$E(h,p)$	Encrypt the hash value h by using private key p by using asymmetric cryptography such as RSA [28], ECC[29], and so on.
$D(s,q)$	Decrypt the digital signature s by using the public key q.

The proposed multi-signature scheme consists of two main phases, signing and verification phase, as described in the following subsection.

1) Signing phase

For instance, there exists a document that requires approval from a total of n parties. The hash value of the document intended for signing is extracted. Subsequently, each signing party contributes the private key necessary for the generation of the digital signature. Every digital signature that is generated will be recorded as a transaction and aggregated into a new block on the blockchain. Finally, the signed document, accompanied by the corresponding public key and the address of the document block, is provided to the intended recipient. The whole process of signing phase is shown in Fig. 5.

Let $U = \{u_1, u_2, \dots, u_n\}$ be a group of n signer and d be the document to signed by all members in U. Every signer in U is required to adhere to the selected asymmetric cryptography protocol. As an example, the cryptography scheme used is RSA [30]. Therefore, each signer has an ordered pair of private key p_i and public key q_i where $1 \leq i \leq n$. The description of the signing phase is as follows.

For all signer u_i in U where $1 \leq i \leq n$ do

- Step 1. Calculate the hash value of the document d.

$$h = hash(d)$$

- Step 2. Encrypt the hash value h by using private key to yield the digital signature s.

$$s_i = E(h, p_i)$$

- Step 3. Every individual digital signature s_i, will be kept as a distinct block within the blockchain. Refer to Fig. 4 in order to determine the type of data that will be stored.
- Step 4. Lastly, send the document d, digital signatures s_i and its addresses in blockchain to the recipient.

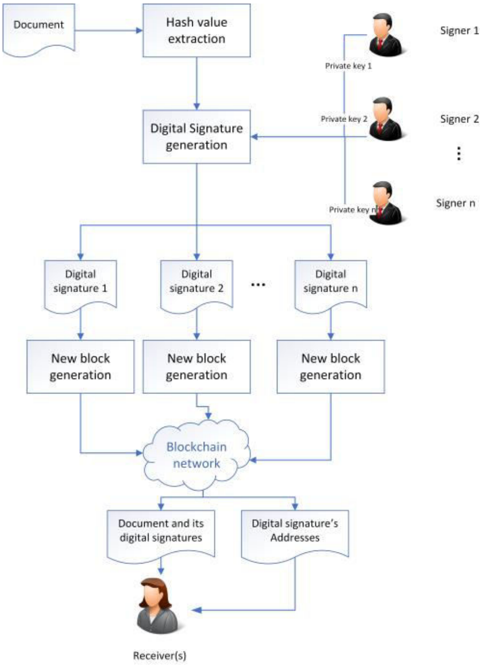


Fig. 5. Proposed signing phase.

2) Verification phase

Once the document and its corresponding digital signature, including the address of the digital signature, have been received, the verification process is initiated. During the verification procedure, the address of the document block requires evaluation. If the block address is deemed authentic, it is necessary to verify the integrity of the digital signature enclosed within the block. The digital signature is decrypted using the public key, and if the resulting value matches the hash value of the document, it can be concluded with certainty that the document has not undergone any modifications. The verification procedure is visually depicted in Fig. 6.

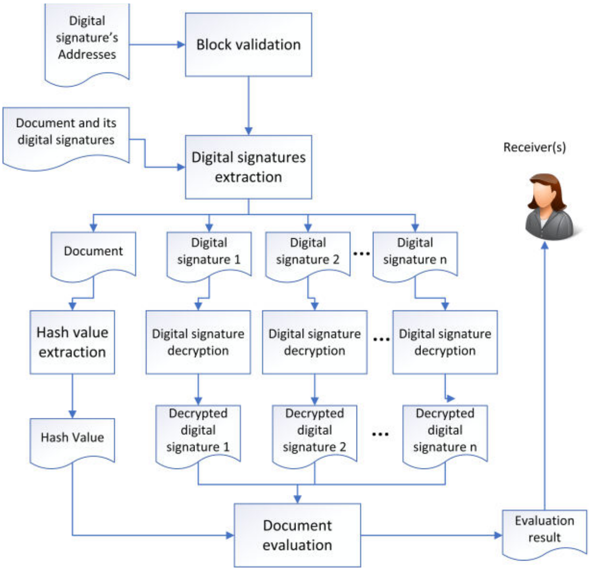

Digital signature's Addresses
Block validation
Document and its digital signatures
Digital signatures extraction
Receiver(s)
Document
Digital signature 1
Digital signature 2
...
Digital signature n
Hash value extraction
Digital signature decryption
Digital signature decryption
...
Digital signature decryption
Hash Value
Decrypted digital signature 1
Decrypted digital signature 2
...
Decrypted digital signature n
Document evaluation
Evaluation result

Fig. 6. Proposed verification phase.

Let $S = \{s_1, s_2, \dots, s_n\}$ be a group of n digital signatures and d be the document to be verified. The decryption process of each digital signature will involve the utilization of the public key q_i belonging to the respective signers in U where $1 \leq i \leq n$. The description of the verification phase is as follows.

For all digital signature s_i in S where $1 \leq i \leq n$ do

- Step 1. Calculate the hash value of the document d.

$$h = hash(d)$$

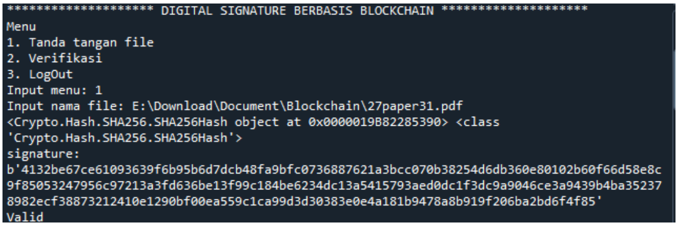
- Step 2. Decrypt the digital signature s_i by using private key of the respective signers.

$$h'_i = D(s_i, q_i)$$

- Step 3. Examine whether h'_i and h are identical. If so, then it can be confirmed that the document is not modified by an unauthorized party.
- Step 4. Display the results of the verification in step 3 to the receiver.

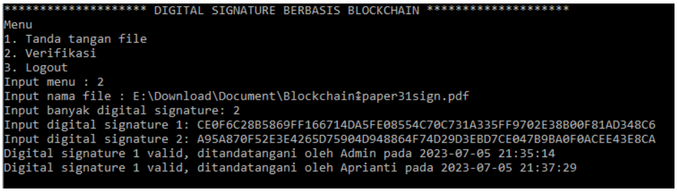
4.2. Prototype Development and Testing

A prototype is developed based on the existing system design. The process of prototyping is conducted with the Python programming language. To access the system, users are required to authenticate themselves by entering a designated username and password. Upon successful authentication with the correct username and password, the user will be presented with the main menu. This menu includes the functionalities of file signature, verification, and logout. In order to apply a digital signature to a file, the user provides the directory address of the file. Subsequently, a digital signature will manifest, as depicted in Fig. 7. While the verification phase is shown in Fig. 8



```
******************** DIGITAL SIGNATURE BERBASIS BLOCKCHAIN ********************
Menu
1. Tanda tangan file
2. Verifikasi
3. LogOut
Input menu: 1
Input nama file: E:\Download\Document\Blockchain\27paper31.pdf
<Crypto.Hash.SHA256.SHA256Hash object at 0x0000019B82285390> <class
'Crypto.Hash.SHA256.SHA256Hash'>
signature:
b'4132be67ce61093639f6b95b6d7dcb48fa9bfc0736887621a3bcc070b38254d6db360e80102b60f66d58e8c
9f85053247956c97213a3fd636be13f99c184be6234dc13a5415793aed0dc1f3dc9a9046ce3a9439b4ba35237
8982ecf38873212410e1290bf00ea559c1ca99d3d30383e0e4a181b9478a8b919f206ba2bd6f4f85'
Valid
```

Fig. 7. Signing a file.



```
******************** DIGITAL SIGNATURE BERBASIS BLOCKCHAIN ********************
Menu
1. Tanda tangan file
2. Verifikasi
3. Logout
Input menu : 2
Input nama file : E:\Download\Document\Blockchain↨paper31sign.pdf
Input banyak digital signature: 2
Input digital signature 1: CE0F6C28B5869FF166714DA5FE08554C70C731A335FF9702E38B00F81AD348C6
Input digital signature 2: A95A870F52E3E4265D75904D948864F74D29D3EBD7CE047B9BA0F0ACEE43E8CA
Digital signature 1 valid, ditandatangani oleh Admin pada 2023-07-05 21:35:14
Digital signature 1 valid, ditandatangani oleh Aprianti pada 2023-07-05 21:37:29
```

Fig. 8. Verify the digital signatures.

Once the prototype has been developed, it requires testing through the black box method to verify its functionality.

Table 3. Test result.

No.	Feature	Operation	Result
1	Login	The user is requested to submit a valid username and password.	Prototype displays the main menu
2		The user is requested to submit a valid username and password.	Prototype displays the warning notification
3	Signing a file/document	The user inputs the document directory with the correct format	Prototype generates a digital signature and stores it in the blockchain
4		The user inputs the document directory with the incorrect format	Prototype displays the warning notification
5	Verify digital signatures	The user inputs the document directory with the correct format without modification within	Prototype display information about the document

No.	Feature	Operation	Result
		the document	signing history
6		The user inputs the document directory with the correct format with modification within the document	Prototype display information that the document is not valid
7		The user inputs the document directory with the incorrect format	Prototype displays the warning notification

5. Conclusion

The test results have successfully met the functional criteria of the system, as outlined in Table 1. The maintenance of data integrity is demonstrated by the presence of operation number 6 in Table 3. The prototype of the proposed method can display a warning message to indicate the invalidity of a modified document when it is submitted by the user. The operational verification of non-repudiation and traceability can be demonstrated through the implementation of operation number 5, as outlined in Table 3. The document's signatory history can be accessed by the user as shown in Fig. 8 and this access can be maintained for an extended period of time.

References

[1] K. Itakura.: "A public-key cryptosystem suitable for digital multisignatures," *NEC J. Res. Dev.*, vol. 71, 1983.

[2] J. Lv, X. Wang, and K. Kim.: "Security of a multisignature scheme for specified group of verifiers," *Appl Math Comput*, vol. 166, no. 1, pp. 58–63, 2005, doi: 10.1016/j.amc.2004.04.108.

[3] S. S. M. Chow, L. C. K. Hui, S. M. Yiu, and K. P. Chow.: "Forward-secure multisignature and blind signature schemes," *Appl Math Comput*, vol. 168, no. 2, pp. 895–908, 2005, doi: 10.1016/j.amc.2004.09.015.

[4] Z. Zhang and G. Xiao.: "New multisignature scheme for specified group of verifiers," *Appl Math Comput*, vol. 157, no. 2, pp. 425–431, 2004, doi: 10.1016/j.amc.2003.08.043.

[5] L. J. Wang and J. J. Chen.: "An improved discrete logarithm-based multisignature scheme," *Security and Communication Networks*, vol. 5, no. 9, pp. 969–973, 2012.

[6] X. Cheng, J. Liu, L. Guo, and X. Wang.: "Identity-based multisignature and aggregate signature schemes from m-torsion groups," *Journal of Electronics (China)*, vol. 23, no. 4, pp. 569–573, 2006, doi: 10.1007/s11767-004-0178-z.

[7] S. Lin, B. Wang, and Z. Li.: "Digital multisignature on the generalized conic curve over Zn," *Comput Secur*, vol. 28, no. 1–2, pp. 100–104, 2009, doi: 10.1016/j.cose.2008.09.002.

[8] Y. S. Chang, T. C. Wu, and S. C. Huang.: "ElGamal-like digital signature and multisignature schemes using self-certified public keys," *Journal of Systems and Software*, vol. 50, no. 2, pp. 99–105, 2000, doi: 10.1016/S0164-1212(99)00080-1.

[9] A. Ansper, A. Buldas, M. Roos, and J. Willemson.: "Efficient long-term validation of digital signatures."

[10] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system.: " *Decentralized business review*, p. 21260, 2008.

[11] M. A. Uddin, A. Stranieri, I. Gondal, and V. Balasubramanian.: "A survey on the adoption of blockchain in IoT: challenges and solutions," *Blockchain: Research and Applications*, vol. 2, no. 2, p. 100006, 2021, doi: 10.1016/j.bcra.2021.100006.

[12] D. Ongaro and J. Ousterhout.: "In search of an understandable consensus algorithm," in *2014 {USENIX} Annual Technical Conference ({USENIX}{ATC} 14)*, 2014, pp. 305–319.

[13] NIST, "FIPS 180-4 Secure Hash Standard (SHS).: " *Gaithersburg, Montgomery County, Maryland: National Institute of Standards and Technology. doi: http://dx. doi. org/10.6028/NIST. FIPS*, pp. 180–184, 2015.

[14] Z. C. Li, J. M. Zhang, J. Luo, W. Song, and Y. Q. Dai.: "Group-oriented (T, n) threshold digital signature schemes with traceable signers," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 2040, no. 004070400, pp. 57–69, 2001, doi: 10.1007/3-540-45415-2_5.

[15] A. Azaria, A. Ekblaw, T. Vieira, and A. Lippman.: "MedRec: Using blockchain for medical data access and permission management," *Proceedings - 2016 2nd International Conference on Open and Big Data, OBD 2016*, pp. 25–30, 2016, doi: 10.1109/OBD.2016.11.

[16] A. Abid, S. Cheikhrouhou, S. Kallel, and M. Jmaiel.: "NovidChain: Blockchain-based privacy-preserving platform for COVID-19 test/vaccine certificates," *Softw Pract Exp*, vol. 52, no. 4, pp. 841–867, 2022, doi: 10.1002/spe.2983.

[17] M. Yusup, Q. Aini, D. Apriani, and P. Nursaputri.: "Pemanfaatan Teknologi Blockchain Pada Program Sertifikasi Dosen," *SENSITIf: Seminar Nasional Sistem Informasi dan Teknologi Informasi*, pp. 365–371, 2019.

[18] I. Permatasari, M. Essaid, H. Kim, and H. Ju.: "Blockchain implementation to verify archives integrity on cilegon E-archive," *Applied Sciences (Switzerland)*, vol. 10, no. 7, 2020, doi: 10.3390/app10072621.

[19] A. Galiev, N. Prokopyev, S. Ishmukhametov, E. Stolov, R. Latypov, and I. Vlasov.: "Archain: A Novel Blockchain Based Archival System," *Proceedings of the 2nd World Conference on Smart Trends in Systems, Security and Sustainability, WorldS4 2018*, pp. 308–312, 2019, doi: 10.1109/WorldS4.2018.8611607.

[20] S. Krishnapriya and G. Sarath.: "Securing Land Registration using Blockchain," *Procedia Comput Sci*, vol. 171, no. 2019, pp. 1708–1715, 2020, doi: 10.1016/j.procs.2020.04.183.

[21] S. Cao, M. Foth, W. Powell, T. Miller, and M. Li.: "A blockchain-based multisignature approach for supply chain governance: A use case from the Australian beef industry," *Blockchain: Research and Applications*, vol. 3, no. 4, p. 100091, 2022, doi: 10.1016/j.bcra.2022.100091.

[22] Y. Yanovich, I. Shiyanov, T. Myaldzin, I. Prokhorov, D. Korepanova, and S. Vorobyov.: "Blockchain-based supply chain for postage stamps," *Informatics*, vol. 5, no. 4, pp. 1–9, 2018, doi: 10.3390/informatics5040042.

[23] J. H. Tseng, Y. C. Liao, B. Chong, and S. W. Liao.: "Governance on the drug supply chain via gcoin blockchain," *Int J Environ Res Public Health*, vol. 15, no. 6, 2018, doi: 10.3390/ijerph15061055.

[24] B. Cook.: "BLOCKCHAIN: TRANSFORMING THE SEAFOOD SUPPLY CHAIN," pp. 1–41.

[25] R. Rivest.: "The MD5 message-digest algorithm," 1992.

[26] D. Eastlake 3rd and P. Jones.: "US secure hash algorithm 1 (SHA1)," 2001.

[27] S. Kerckhof, F. Durvaux, N. Veyrat-Charvillon, F. Regazzoni, G. M. De Dormale, and F. X. Standaert, "Compact FPGA implementations of the five SHA-3 finalists," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7079 LNCS, pp. 217–233, 2011, doi: 10.1007/978-3-642-27257-8_14.

[28] R. L. Rivest, A. Shamir, and L. Adleman.: "A method for obtaining digital signatures and public-key cryptosystems," *Commun ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[29] N. Koblitz, A. Menezes, and S. Vanstone.: "The State of Elliptic Curve Cryptography," 2000.

[30] R. L. Rivest, A. Shamir, and L. Adleman.: "A method for obtaining digital signatures and public-key cryptosystems," *Commun ACM*, vol. 21, no. 2, pp. 120–126, 1978.