Journal of Information Technology Management

Print ISSN: 2008-5893
Online ISSN: 2423-5059

# Blockchain and Bigdata to Secure Data Using Hash and Salt Techniques

**Fudhah A. AlSelami***

*Corresponding Author, Department of Management Information Systems, College of Business, University of Jeddah, Al-kamel Governorate Branch, Jeddah, Saudi Arabia. E-mail: falsulami@uj.edu.sa

## Abstract

In recent times amount of data is increasing rapidly and analysis of data is a must to come up with business decisions, predictions etc. It’s not just text are numbers which has to be stored properly. Data Types these days varies from images, video, social media data, surveys to blogs etc. When this is the case its mandatory to deal with Bigdata and to safeguard those Bigdata. Technologies available in Bigdata and Bitcoin helps us in achieving this. Bigdata technologies helps in storing the unstructured data effectively and processing of such voluminous data is taken care of. Hashing algorithms are used in Blockchain to keep the data safe. Hashing algorithms like SHA 256 are used to make it more secure. Its almost not possible to break the data available in Blockchain. Let’s try to secure the data even more using encryption algorithms to make sure that proper data is used for analysis etc.

**Keywords:** Blockchain, Bigdata, Salt Techniques, Data Using Hash.

Journal of Information Technology Management, 2022, Vol. 14, No.2, pp. 15-25
Published by University of Tehran, Faculty of Management
doi: https://doi.org/ 10.22059/JITM.2022.86924
Article Type: Research Paper

Received: October 26, 2021
Received in revised form: February 07, 2022
Accepted: March 18, 2022
Published online: April 20, 2022

## Introduction

The growth of voluminous amount of data has led to less use of traditional storage and processing methods. Using of RDBMS in the current environment of unstructured or semi structured data has become less possible. Technologies like Hadoop that support storage of data of various type are becoming mandatory. Hadoop uses HDFS file storage method. Here raw data is stored and only when its needed processing is done to unstructured data.

Blockchain technologies also let data to be stored in blocks in an effective manner. And as it uses hashing techniques, data is quite safe here. Hash functions create fixed length output (Baygin, N., et al. 2019).

## Traditional RDBMS

RDBMS stands for Relational Database Management Systems. Data here are stored in data warehouses as Tables. Tables consists of data. Tables are of rows and columns and mostly they have numbers and text as data.

Table 1. Sample Table in RDBMS

| ID | NAME | AGE | OCC |
|---|---|---|---|
| 001 | X | 22 | Business |
| 002 | Y | 23 | Teacher |
| 003 | Z | 24 | Doctor |
| 004 | Z1 | 30 | Lawyer |
| 005 | Z2 | 32 | Fashion designer |

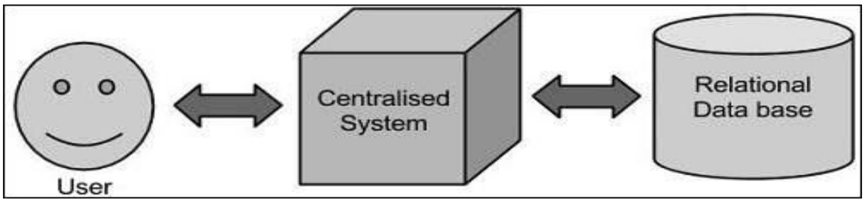


Fig 1. Traditional way

## Bigdata Technologies

Bigdata is about large voluminous data. Bigdata is of different data types, for e.g., texts, numbers, audio, video, social-media contents, blogs, surveys etc.… For greater variety, volumes and velocity of data bigdata technologies are useful. In this sort of scenarios, traditional methods find it difficult to manage. In the case of Hadoop environment data sets are distributed and file system used is HDFS (Hadoop Distributed File System).

Storage of data is taken care of HDFS. HDFS takes replicas of data and distributes them. The master Node takes care of parallel processing of data using Map reduce. In the Slave Nodes data are present.

## HDFS

HDFS stands for Hadoop Distributed File System. In Hadoop file system, even GB to TB of data can be stored. Data is divided and made available in n no. of nodes. To make sure that data is not lost replicates of data is made available in servers. Even on failure in one server, data can be fetched from another server.

HDFS can be used when we are dealing with a larger file which is not manageable in a traditional way. And HDFS are built on low-cost Hardware. To access the first data, it takes time in HDFS.

The name node acts as a master. It manages the metadata of all the files. As multiple clients can work parallelly, this single machine manages all the tasks needed to be done.

Data Node is used to store the data and retrieve them when needed.

**Mapreduce**

This module helps in parallel processing. Speed is quite high. And the data is finally combined from many servers and the output is generated. All forms of data can be stored using Hadoop e.g., Data from web, social networks, images, video etc. (Acharjya, D. P., & Ahmed, K. 2016).

Mapper class takes the input, tokenizes it, maps and sorts it. The output of Mapper class is used as input by Reducer class, which in turn searches matching pairs and reduces them.

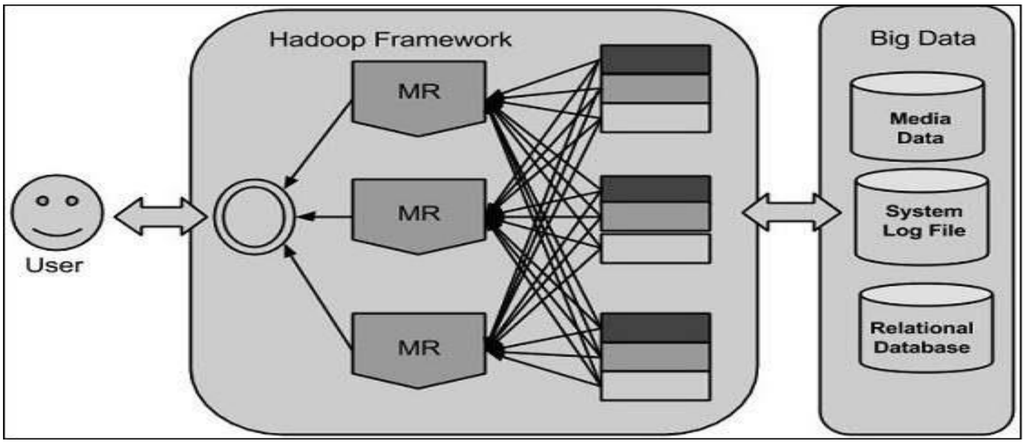


Fig 2. Big Data Technologies

**Characteristics of Big Data**

3 main Characteristics of Big Data are

1.Volume (Mukherjee, S., & Shaw, R. 2016).

Potential insights can be achieved only using large volume of useful data. In recent times most of the data are of unstructured or semi structured forms. There are Penta bytes of Data. File structures like HDFS helps in saving large volume of data.

2.Variety (Narayanan, U., et al. 2017)

RDBMS are good at handling structured data. But when it comes to unstructured form storing and processing becomes tedious. Data forms include textual data, audio forms, video forms, data from social media, surveys, email messages, newspaper articles, blog posts etc.

3.Velocity

At a greater speed data are generated and they must be processed at a greater speed too. As big data are more of real time data, mostly they are often.

## Blockchain

Block chain consists of n number of blocks and nodes. Data resides in the block. Nonce and Hash value is created for each node. The hash value makes sure that the data is safe inside the block. Decentralization approach is followed in Blockchain (Zheng, Z et al. 2017).

Blockchains are used in applications like IOT, Healthcare, Government sectors, Finance, cryptocurrency exchange, music royalties tracking etc.

## Types of Blockchain

1.Public Blockchain: When a new block is created, anyone can add it to their blockchain after checking it for any tampering.

2.Private Blockchain: Private – Its for an organization. Within an organization anyone can add after checking it.

## Encryption and Decryption

To make sure that data is generally safe in Network, encryption algorithms are used. Encryption algorithm changes plain text into cipher text. This cipher text makes sure that the original data cannot be read by a 3rd person. To get back the original text, decryption is performed.

## Sample Encryption Algorithms:

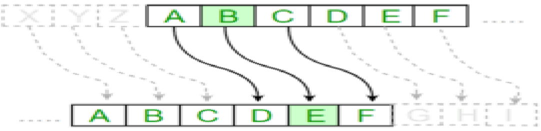
1. Triple DES
2. RSA
3. AES etc…

## Types

1. Substitution Ciphers

Characters from plain text are substituted by some other character. Thereby can come up with a encrypted text.

Eg.l



2. Transposition Ciphers

E.g., Columnar Transposition. Here plain text is written horizontally and read vertically and it becomes the cipher text.

| | | |
|---|---|---|
| H | E | L |
| L | O | H |
| A | I | |

Cipher Text: hlaeoilh

**Two main forms of Encryption**

1. Symmetric Encryption

It is the commonly used one. Here there is only 1 key involved. Single key is used to encrypt the data as well as to decrypt the data. Sender and receiver use the same key here.

E.g., for Symmetric Encryption:

- Data Encryption Standard (DES)
- Triple Data Encryption Standard (Triple DES)
- Advanced Encryption Standard (AES)
- International Data Encryption Algorithm (IDEA) etc.

Advantages:

- Speed
- Security – Standard algorithm takes longer time to be cracked.
- Industry adoption and acceptance

Drawback:

- The key must be transmitted, and this might add up to the vulnerability.

2. Asymmetric Encryption

Also known as public key encryption. Here key pairs are used. Public key and Private key. Public key is used for encryption part and private key is used for decryption part.

As the private key is kept safe, unauthenticated access can be minimized. As the hacker might not be aware of private key, he will not be able to decrypt the data.

E.g., for Asymmetric Encryption:

- Rivest Shamir Adleman (RSA)
- Digital Signature Standard (DSS)
- Elliptical Curve Cryptography (ECC)
- Diffie-Hellman Exchange Method etc.

Advantages

- Key distribution not necessary
- Exchange of private keys not necessary

Drawbacks

- Slower
- Complex

## Methodology

Blockchain is simply a chain of blocks which can hold data (Zheng, Z et al. 2017). E.g., A block with details like From, To, amount etc. Each block consists of hash value of the earlier block too. If someone makes modifications to a block, it an identified from the next block too. So, its difficult for an intruder to affect a block. Plus, hashing algorithms like SHA 256 are so powerful that security is enhanced using those hashing algorithms.

| | |
|---|---|
| Block 1<br>Hash value<br>Previous Hash Value | Block 2<br>Hash value<br>Previous Hash value |

For e.g. If an attacker changes data of Block10, Hash value of Block10 changes in Block 10 and the previous Hash value of Block 11 will contain the old Hash value. This makes this method so strong.

### Proof of work

In today's world machines are so powerful hence an attacker might make changes to a block and can compute Hash values accordingly in a fraction of seconds. To avoid this proof of work concept is introduced. For e.g., Proof of work might take a few minutes.

### Distributed Approach

When a new block is created, it is sent to all of them of the Network. They check the blocks for any tampering and finally add the block to their blockchain.

### Security Enhancements using Hashing Technique

Hashing techniques helps enhancing security features. Hashing is different from encryption techniques. In the case of encryption, it's possible to decrypt and get back the original text. But in the case of Hashing, hash values are used to ensure that the data are not altered.

SHA is a famous hashing technique. The size of the output can be 256/512 bits.

STEP 1: Pre-processing work. Represent the message in binary and add zero's + 64 bits so that the data is of multiples 512 bits.

STEP 2: Initialize 8 Hash values, 8 buffers a to h.

STEP 3: Initialize Round Constant k of 64 bits.

STPE 4: Divide the entire message block into n chunks of 512 bits

STEP 5: Perform 64 rounds of operation to those 512 bits. 1st round results will act as input for the next round and so on.

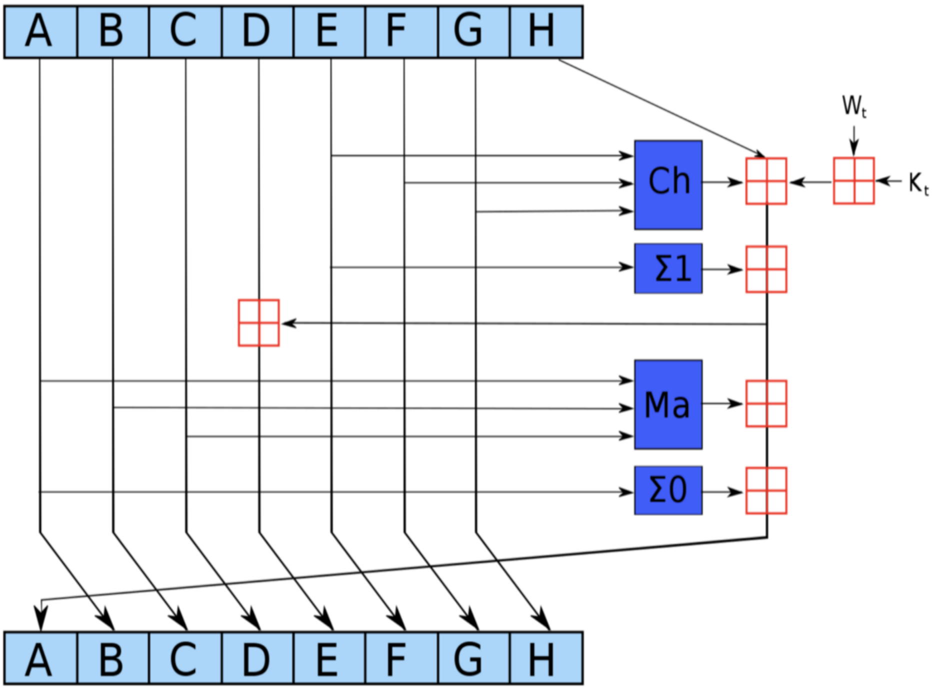
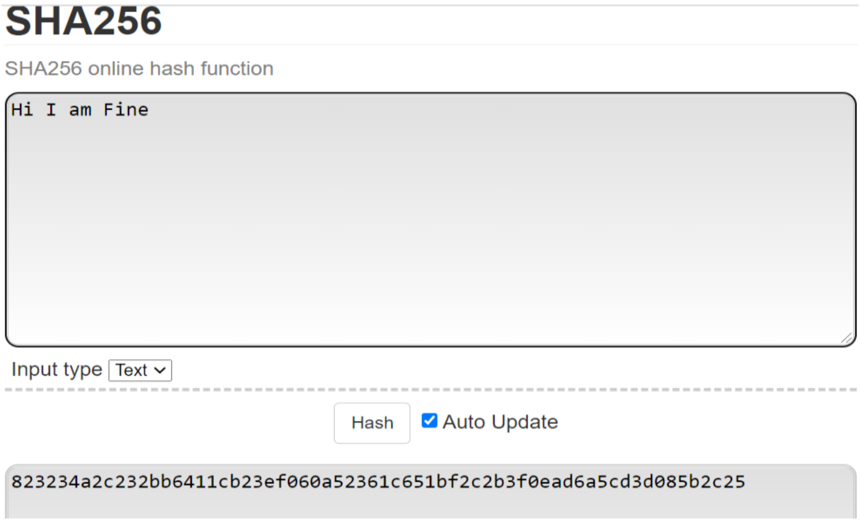


Fig 3. Single round of SHA 256 Algorithm

STEP 5: The result of the 64th round is the final output. And the final hash value will be of 256 Bits.

The 64 rounds make the hashing algorithm so strong. But the technologies and computing power available these days make it possible to find the Hash value.

As an additional security measure Proof of work technique is used in blockchain techniques. And as hash values of previous blocks are available in the current block, it helps to find the tampering.



```
Hi I am Fine
```

Input type: Text

☑ Auto Update

```
823234a2c232bb6411cb23ef060a52361c651bf2c2b3f0ead6a5cd3d085b2c25
```

Fig 4. Hash Value for an input.

### SALT value

Salt value helps to increase the uniqueness in Hashing Technique. A notable part in hashing algorithm is that for the same input it produces the same output. Though Hashing algorithms are strong, computing power is also strong these days. So, to make things even more difficult for the attacker/hacker Salt value can be used.

Adding salt to the hashing technique makes sure that the outputs are unique irrespective of the input as salt value is added to the input. This helps us to face attacks like hash table attacks, dictionary attacks, brute-force attacks etc.

E.g., Let us consider a Scenario where 2 people are considering same password.

Password of A: Kitty

Password of B: 12345

Password of C: Kitty

Password of D: x123

After applying SHA 256 Algorithm, this will be the following output:

Hashed Value for A:
67731ff58137eb39713ae30eba33c54c8c1d5418e081428ca815e4e733d64f6d

Hashed Value for B:
5994471abb01112afcc18159f6cc74b4f511b99806da59b3caf5a9c173cacfc5

Hashed Value for C:
67731ff58137eb39713ae30eba33c54c8c1d5418e081428ca815e4e733d64f6d

Hashed Value for D:
a166ea7aef8fa5172bb82006d309b9b713f035a569525c1e16a43e7b28b1b289

Hash value of A and C are similar as the input data is the same. These sorts of things might help the attacker to trace the original data. In the above case if the password of A is known by the attacker, he can very well know the password of C too. To avoid these sort of scenarios Salt method can be used.

We can either append the salt value or we can prepend the salt value.

For e.g., Password + Salt value or salt value + Password.

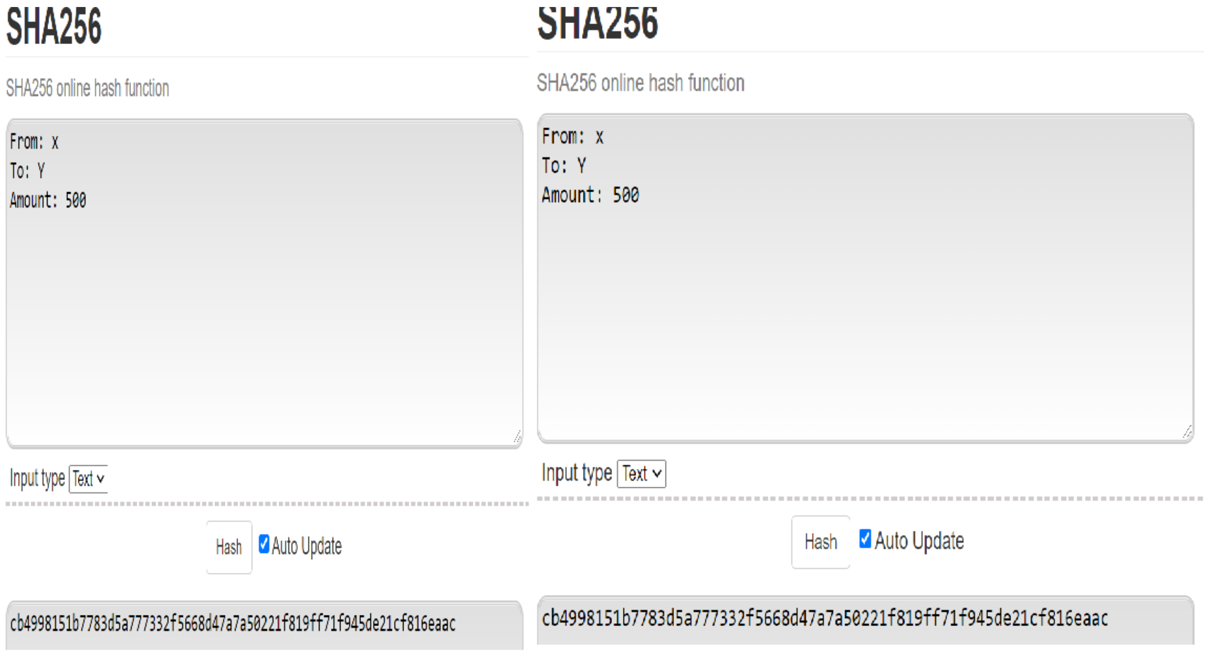
Salt Value: 123. After prepending salt value to the password, we get 123kitty.

And the respective hash value would be:
ac80cf42139999bd5968ded4acacc421089f65ef5049fd471e52d895cf558db1

We must also make sure that we are adding a cryptographically strong salt.

## Results

After applying Hash functions to two blocks in blockchain where the block contents are similar.



```
From: x
To: Y
Amount: 500
```

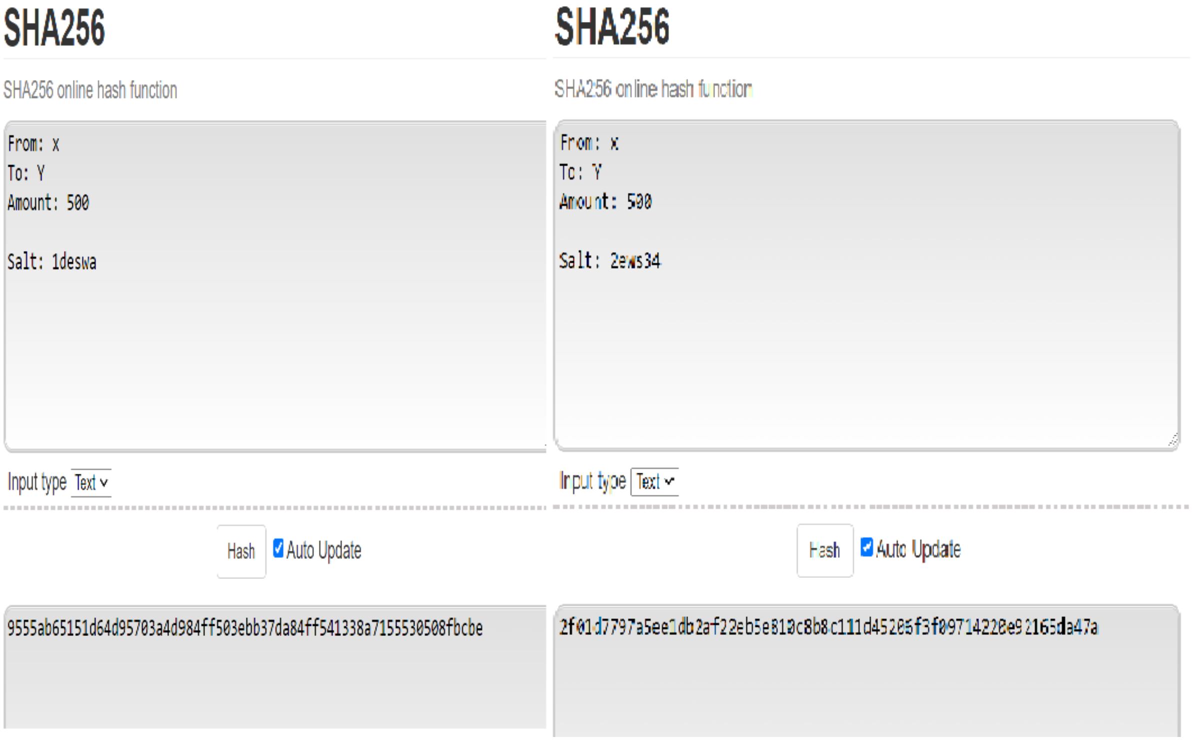
```
cb4998151b7783d5a777332f5668d47a7a50221f819ff71f945de21cf816eaac
```

```
From: x
To: Y
Amount: 500
```

```
cb4998151b7783d5a777332f5668d47a7a50221f819ff71f945de21cf816eaac
```

Fig 5. Hashed value for blocks without salt value

After appending salt to the block data in the blockchain.



```
From: x
To: Y
Amount: 500

Salt: 1deswa
```

```
9555ab65151d64d95703a4d984ff503ebb37da84ff541338a7155530508fbcbe
```

```
From: x
To: Y
Amount: 500

Salt: 2ews34
```

```
2f01d7797a5ee1db2af22eb5e810c8b8c111d45205f3f09714220e92165da47a
```

Fig 6. After adding salt value to the block contents, where the block contents are the same.

## Conclusion

Big data and Blockchain technologies are supportive to handle the large voluminous data and they make sure that the data is secure. Hashing algorithms like SHA 256 are used to make sure that data is safe. But the problem with hashing techniques is that for the same input, same output is always produced. This might act as a clue to the hackers. To sort this out salt techniques are used to the hashing functions. Thereby data can be even more secure.

## Future Enhancements

As various attacks are possible, solutions to handle them can be found. E.g., Routing attack, 51% attacks. In 51% attack, if an entity can control 51% or more of the network nodes, then it can result in control of the network.

## Conflict of interest

The authors declare no potential conflict of interest regarding the publication of this work. In addition, the ethical issues including plagiarism, informed consent, misconduct, data fabrication and, or falsification, double publication and, or submission, and redundancy have been completely witnessed by the authors.

## Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article

## References

Acharjya, D. P., & Ahmed, K. (2016). A survey on big data analytics: challenges, open research issues and tools. International Journal of Advanced Computer Science and Applications, 7(2), 511-518.

Baygin, N., Baygin, M., & Karakose, M. (2019). Blockchain Technology: Applications, Benefits and Challenges. In 2019 1st International Informatics and Software Engineering Conference (UBMYK) (pp. 1-5). IEEE.

https://www.tutorialspoint.com/hadoop/hadoop_big_data_solutions.htm

https://www.guru99.com/blockchain-tutorial.html

https://infosecwriteups.com/breaking-down-sha-256-algorithm-2ce61d86f7a3

https://emn178.github.io/online-tools/sha256.html

https://auth0.com/blog/adding-salt-to-hashing-a-better-way-to-store-passwords/

https://www.trentonsystems.com/blog/symmetric-vs-asymmetric-encryption

Mukherjee, S., & Shaw, R. (2016). Big data–concepts, applications, challenges and future scope. International Journal of Advanced Research in Computer and Communication Engineering, 5(2), 66-74.

Narayanan, U., Paul, V., & Joseph, S. (2017). Different analytical techniques for big data analysis: A review. In 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS) (pp. 372-382). IEEE.4.

Zheng, Z., Xie, S., Dai, H., Chen, X., & Wang, H. (2017). An overview of blockchain technology: Architecture, consensus, and future trends. In 2017 IEEE international congress on big data (BigData congress) (pp. 557-564). IEEE.

**Bibliographic information of this paper for citing:**

AlSelami, Fudhah A. (2022). Blockchain and Bigdata to Secure Data Using Hash and Salt Techniques. *Journal of Information Technology Management*, 14 (2), 15-25. https://doi.org/10.22059/jitm.2022.86924