# BLOCKCHAIN BASED SECURE DATA TRANSFER

# IN CLOUD STORAGE

Dr. Sangeetha Kalyaniraman(1), Abitha U(2), Kaviasree S(3), Phooja Shree R(4)

*(1)Assistant Professor, Department of CSE, Panimalar Engineering College, Chennai-600123, India*

*(2,3,4)Student, Department of CSE, Panimalar Engineering College, Chennai-600123, India*

Mail: [sangeethakalyaniraman@gmail.com](mailto:sangeethakalyaniraman@gmail.com) , [*abiudaya02@gmail.com*](mailto:abiudaya02@gmail.com%20) *,* [*srikavia219@gmail.com*](mailto:srikavia219@gmail.com) *,* [*phoojashree123@gmail.com*](mailto:phoojashree123@gmail.com)

***ABSTRACT:*** *The growth in technological advancements has led to the abrupt rise in the improvement of digital belongings and transactions. The documents or files are more prone to information breaches and theft from the intruders who are successful in obtaining the data. This led to the advancement of adopting concepts from the blockchain. Blockchain is best known as data sharing technology with respect to its application in Organization data transactions. A blockchain is a growing list of distributed records that are connected to each other through the usage of cryptography. It offers data security in an encrypted format.*

***KEYWORDS:*** *Blockchain, Data transfer, Cloud Storage, SHA-256, AES*

# INTRODUCTION

# The amount of data in today's eco-system is growing tremendously. Data is essential for every organization in the modern world, and it is frequently collected and evaluated, leading to technological improvements and economic prosperity for nations. The information gathered helps businesses and organizations reduce expenses and improve key operations so they can compete with others. Data is a valuable asset for every organization in the world. In today's data-driven society, user privacy is a major public concern that needs to be properly taken into account. To ensure the safety measures against the security attack blockchain is used. Taking into account the case of a company or an organization, the confidential files are first converted to fixed-sized blocks of data which are provided primary encryption using modified AES and SHA-256. IP address is used in such a way where for each action it is verified. Recent years have seen a sharp rise in demand for cloud storage. Despite the many advantages of cloud storage, like its convenience and adaptability, customers typically have no idea where their data is truly kept. Due to this constraint, customers may feel less secure and less motivated to trust the storage provider, or the cloud may even be unsuitable for keeping data that must adhere to tight geographic restrictions. LAST-HDFS, an existing solution that incorporates Location Aware Storage Technique (LAST) into the open-source and free Hadoop distributed file system seeks to figure out this problem (HDFS). The LAST-HDFS system imposes location-aware file allocations and actively monitors file transfers in order to detect potentially illicit transmissions in the cloud. "Illegal transfers" in this system refer to attempts to transfer private information without following the file owner's and its rules' ("legal") rules. The algorithms maximize the probability of keeping data items with similar personal requirements in the same area and model file transfers over nodes as a weighted graph. Users can keep track of the real-time communication taking place between cloud nodes on each individual node with the aid of a socket monitor that is included.

# PROPOSED SYSTEM

The Proposed System considers an organization into account where the input file or data which a project manager or any other employee in a particular organization needs to share with others in the organization is converted into fixed-sized blocks of data, which is apparently similar to the concept used in blockchain. Here the manager browses and adds different files for each section of employees with different access rights, so the employee with the respective access right (valid organization IP address) could request for the file and with the valid secret key the employee can access the entire file. The approach not only ensures confidentiality and integrity of files, but also provides a scalable key management mechanism for file sharing among multiple users. In the proposed work when an unknown user make the request, they could not access the data, it will show empty or null value in authorized panel services. After checking the requests by authority person, if the user is valid, then the user can get the response through mail and if the user is not valid then their request will be rejected. The Blockchain algorithm we use here is SHA-256 and AES for encryption and decryption.

# 3. ALGORITHM DESCRIPTION

* 1. **AES (Advanced Encryption Standard):**

The AES Encryption algorithm, also referred to as the Rijndael algorithm, is a symmetric block cypher algorithm with a 128-bit block/chunk size. It uses keys of 128, 192, and 256 bits to convert each of these discrete blocks. In order to create the cipher text, it encrypts the blocks and combines them. A substitution-permutation network, or SP network, is the foundation of this system. It involves a number of interconnected processes, some of which include bit shuffles and others which involve substitutions, which are the act of changing inputs into particular outputs (permutations). When sending files in an encrypted manner between colleagues, AES is employed. Blocks of 128 bits each are used for AES encryption of data. Based on the length of the key, the number of rounds will vary as follows:

1. For 128bit key there are 10 rounds

2. For 192bit key there are 12 rounds

3. For 256bit key there are 14 rounds

**4.1.1. ENCRYPTION:**

Each round comprises of 4 steps:

1. Sub Bytes
2. Shift Rows
3. Mix Columns
4. Add Round Key



Encryption Process

**SUB BYTES:**

At this stage, one byte is replaced with another. It is carried out utilizing a lookup table also known as the S-box. The state array's byte a(i,j) is replaced with a Sub Byte S(a(i,j)) in this Sub Byte step. In order to replace the sixteen input bytes, a constant table (S-box) provided in the design is looked up. The resultant is in a form of matrix with 4 rows and 4 columns.

**SHIFT ROWS:**

The matrix's four rows are all moved to the left. The right side of the row is used to re-insert any entries that "slip off." This is how shift is done:

1. The top row is not moved.

2. The second row has been moved left one (byte) place.

3. The third row has been moved two spaces to the left.

4. The fourth row has been moved three spaces to the left.

The outcome is a new matrix made up of the same 16 bytes but with their positions altered.

**MIXCOLUMNS:**

A unique mathematical function is used to change each column of four bytes. The four bytes of one column are entered by this function, which returns four entirely new bytes that replace the original column. The outcome would be another new matrix with 16 more bytes. It must be noted that the final round does not include this phase.

**ADDROUNDKEY:**

The round key's 128 bits are now XORed (XOR Operation) with the matrix's 16 bytes, which are now addressed as 128 bits. If this is the final round, the cypher text will be generated. If not, the generated 128 bits are converted into 16 bytes, and the procedure is repeated.

**4.1.2. DECRYPTION:**

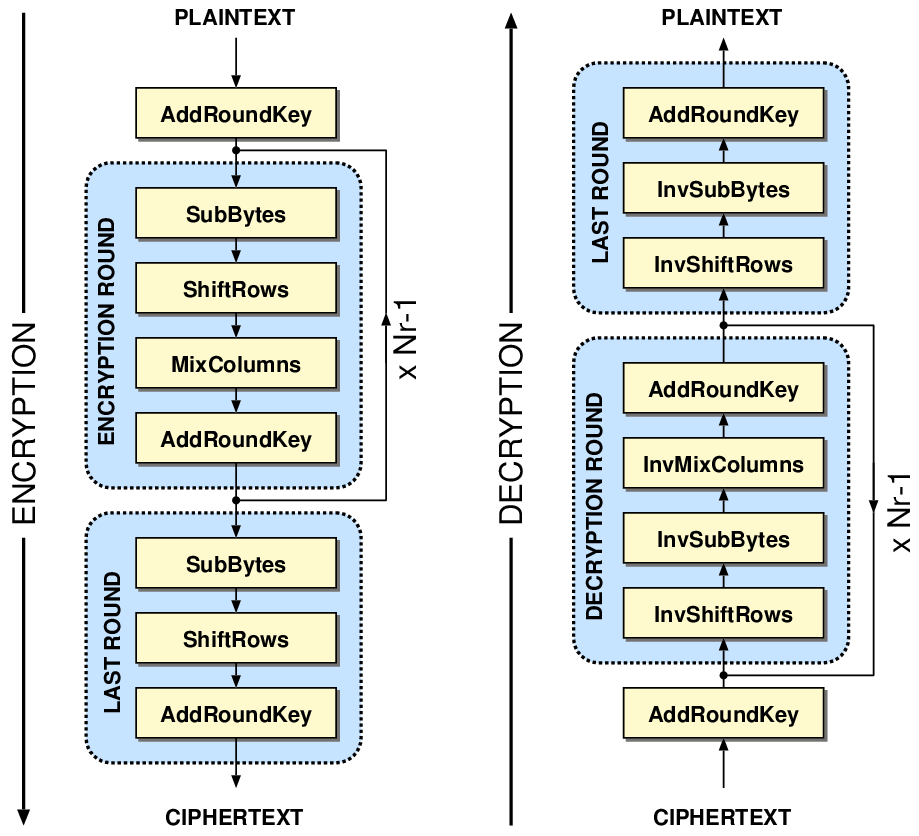
An AES cipher text's decryption procedure is quite identical to its encryption procedure in reverse. The four stages are carried out in reverse order with each round.

1. Add round key

2. Mix columns

3. Shift rows

4. Byte substitution

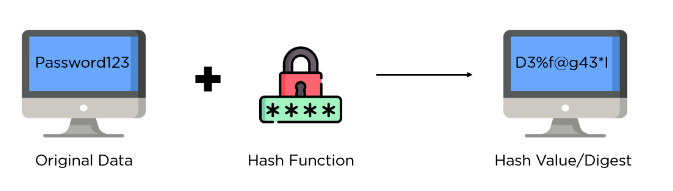


* 1. **SHA-256 (Secure Hash Algorithm – 256):**

Secure Hash Algorithm, or SHA, is the name of the family of algorithms that includes SHA 256. The 256 in the name indicates that no matter how large the plaintext or clear text is, the hash result will always be 256 bits. During the execution of the SHA-256 algorithm, a piece of data is transmitted via a function that applies mathematical operations to the plaintext. The hash function produces the hash value/digest, which is referred to as its output. In contrast to certain other well-known hashing algorithms, SHA-256 is secure and has not been "broken," which is the main reason why technology employs it. Also, it has no known flaws that would make it unstable. The hash value is always 256 bits no matter how big the clear text or plaintext is. The number 256, which is used in the name to denote the very last hash digest value, has this meaning.

**HASHING:**

A hash value, also known as a hash function, is produced by a mathematical operation and an arbitrary-sized file, such as an email, document, picture, or other type of data. As a one-way function that is specific to the file being hashed, a derived hash cannot be reversed to find more files that would provide the same hash result.



**CHARACTERISTICS OF SHA-256 ALGORITHM:**

## sha_chars.

* Message Length: The clear text's length should be no more than 264 bits. To maintain the highest level of randomness in the digest, the size has to fit the comparison area.
* Digest Length: The hash digest for the SHA-256 algorithm must be 256 bits long, for the SHA-512 algorithm 512 bits, etc. Huge digests often represent a large increase in calculations at the expense of storage and efficiency.
* Irreversible: Every hash function, including SHA 256, is irreversible by design. If you have created the digest already, neither you nor the digest should receive the original cost when you run it by the hash function once more.

**STEPS IN SHA-256 ALGORITHM:**

1. **PADDING BITS:**

In order to make our original message's length match the minimum length needed for the hash function, the first step of the hashing function is attaching bits to it. The message is given a few more bits, leaving the length exactly 64 bits shy of a factor of 512. The initial bit of the addition should be a one, and the following bits should all be zeros.

M + P + 64 = n x 512

(i. e) M = length of original message  
 P = padded bits

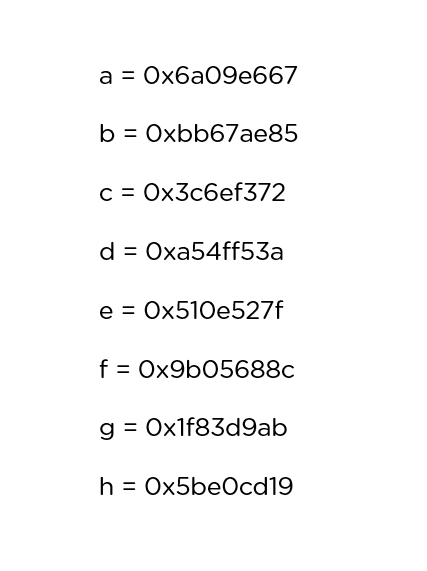
n = constant

1. **PADDING LENGTH:**

The resultant plaintext is then multiplied by 64 bits to make it a factor of 512. Using the modulus of the original clear text without the padding, the 64 bits of characters are calculated.

1. **INITIALIZE BUFFER:**

A 64-key array (K[0] to K[63]) with eight buffers each holding a default value is initialized.



1. **COMPRESSION FUNCTION:**

The algorithm divides the whole message into many blocks, each of 512 bits. The output of each block serves as the input for the subsequent block during each block's 64 rounds of execution. As a result, after each iteration, the block's final output is used as the input for the subsequent block. When you get to the final 512-bit block, you consider the output of the entire cycle to be the overall hash digest. According to the name of this algorithm, this digest will have a length of 256 bits.

**APPLICATION OF SHA-256:**

1. Digital Signature Verification
2. Password Hashing
3. SSL Handshake
4. Integrity Checks

# 4. CONCLUSION

# Several sectors, including machine learning, banking, and the military, are utilizing blockchain, a new technology. It also has some restrictions, such as immutability for some applications where information needs to be altered. Despite these limitations, we must accept and promote its widespread adoption across a range of industries. In this paper, we offered a more secure way to share files and store them on a reliable platform like the blockchain. This technique can partially eliminate all the drawbacks of having a centralized network for sharing confidential files within an organization.

# REFERENCES

[1] Adam Bowers, Cong Liao, Douglas Steiert , Dan Lin , Anna Squicciarini, and Ali Hurson, “Detecting Suspicious File Migration or Replication in the Cloud”, 2021

[2] Anusree K, Jagan Sathiaseelan Vadekkat, Abhinu R Dev, Abhinav, “Decentralized File Transfer System Blockchain-based File Transfer”, 2022

[3] Devesh Solanki , Dev Mehta , Kaushal Modani, Sashank R M Dr Rashmi S, “An Android File Sharing System Using BlockChain & IPFS”, 2021

[4] Jin Sun, Xiaomin Yao , Shangping Wang , and Ying Wu, “Blockchain-Based Secure Storage and Access Scheme For Electronic Medical Records in IPFS”, 2020

[5] Anjaly Renny , Anjana Santhosh , Neeraja S, Teena Noyal, “File Storing and Sharing using Blockchain”, 2020

[6] Shujie Cui, Muhammad Rizwan Asghar, Giovanni Russello, ” Towards Blockchain - based Scalable and Trustworthy File Sharing”, 2018

[7] Xiaochen Zheng, Raghava Rao Mukkamala, Ravi Vatrapu, Joaqun Ordieres-Mere, “Blockchain-based Personal Health Data Sharing System Using Cloud Storage”, 2018

[8] Mathis Steichen, Beltran Fiz, Robert Norvill, Wazen Shbair and Radu, “Blockchain-Based, Decentralized Access Control for IPFS”, 2018

[9] Hoang Giang Do, Wee Keong Ng, “Blockchain-based System for Secure Data Storage with Private Keyword Search”, 2017

[10] Marlene Goncalves, Graciela Perera and Stephen Rodabaugh, “Detecting illegal file sharing in Peer-to-Peer networks using fuzzy queries”, 2010