International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 08 | Aug 2020 www.irjet.net p-ISSN: 2395-0072

BLOCKCHAIN FOR CONTROLLING SOCIAL NETWORK ACTIVITIES AND AUTHORIZATION PROCEDURES

Kathirvel.P[1], Ramesh.C[2]

ABSTRACT: The main of this project is to develop a Blockchain working model for internal security in Social Network Environment with multiple users Interface model. In this project, we are using a SHA256 hash algorithm for the transaction and creating hash blocks for each user. Here in this application, we will generate encoding cryptographic key for viewing the communication and transaction between multiple users. The key is created by using the Rijndael cryptographic algorithm. The primary working is Stop spreading fake and old messages around a social network and Identifying false and irreverent Message or news spreaders. In case of any changes done in the middle node, using the hash function and SHA 256 algorithm, the man in the middle can be identified. The entire hash should be exactly matched with the first hash. So that data cannot be changed in the middle. In case of any changes occurred, the data owner will get noticed.

Keywords: Blockchain, cryptographic algorithm, Network,

I. INTRODUCTION

Social networking on social media platforms includes using the internet to link users to colleagues, family and acquaintances. Websites and social media aren't just about meeting new people online,but that happens.Rather, they're mainly about interacting with your real- life colleagues, relatives and acquainting.

Creating social networking management technology using blockchain [1]. This paper presents a model in which a social networking management model can be constructed into a private blockchain to manage the history of social networking efficiently.

Microblogging is a web service that allows the subscriber to broadcast short messages to other subscribers of the service. Micro articles may be made freely accessible on aWeb site and circulated to a limited user party. Subscribers may read web micro blogposts or order that alerts be sent as an instant message to their laptop in live time, or submitted through one platform to another. Considering the smaller amount of time and effort to make a post this way or share an update, microblogging has the potential to become a new, informal communication medium, especially for collaborative work within organizations.

II. BLOCKCHAIN

Blockchain is a common term which is used in emerging developments in technology. Currently used in terms of Cryptocurrencies. Bitcoin, link Ethereum any applications which can be optimized through decentralization and which need to be highly protected could opt for this technology[3]. Blockchain technology has already embedded its unique transparency, security and versatility in the finance and banking domains.

This paper provides a description of key characteristics, architecture, and taxonomy of blockchain technology. Moreover, the paper provides an insight into the popular consensus algorithms, technical challenges, and major application areas. Future trends and signs of progress in the blockchain technology were discussed [4]. The blockchain provides the integrity of the data using the mesh tree of the structure shown in Figure 1 below. When a new block is created, the block contains a hash of the previous transaction history in the header. These structures allow the blockchain to verify the gastro-modulation of data and provide the integrity of the data.

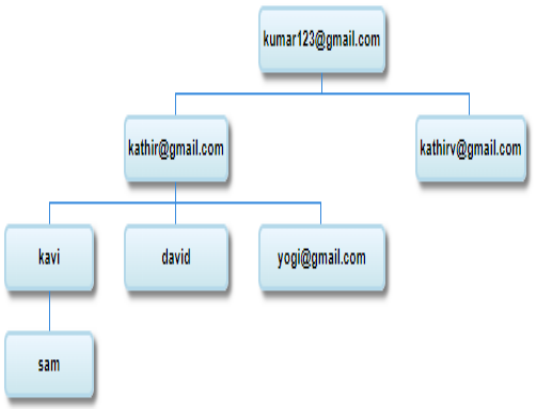

Figure 1. Sending Message in blockchain Tree

III. HASHING ALGORITHM

Hashing algorithms are just as abundant as encryption algorithms, but there are a few that are used more often than others. Some common hashing algorithms include MD5, SHA-1, SHA-2, NTLM, and LANMAN.

A. MD5

This is the fifth edition of the Message Digest algorithm. MD5 produces outputs which are 128-bit. MD5 was a hashing algorithm which was very widely used. This was before flaws started to emerge in the algorithm. Many of those vulnerabilities were expressed

as collisions. MD5 started to be phased out because of this. The simple prerequisite of any cryptographic hash func on is that it should be mathematically impossible to locate two distinct texts but have the same meaning. MD5 fails this criterion catastrophically; on an ordinary home computer, these collisions can be detected in seconds

B. SHA-1

This is the second edition of the Secure Hash Algorithm, with the first being SHA-0. SHA-1 produces outputs of 160bits. SHA-1 is one of the key algorithms that started replacing MD5 after vulnerabilities were discovered. SHA-1 achieved widespread recognition and use. In addition, SHA-1 was designated as a hashing algorithm compliant to FIPS 140.

C.SHA-2

In reality this is a suite of hashing algorithms. The Suite includes SHA-224, SHA-256, SHA-384, SHA-512. Every algorithm displays the length of its output. SHA-2 algorithms are safer than SHA-1 algorithms but SHA-2 has not been used widely.

D.NTLM

It is an algorithm for the NT LAN Manager algorithm. During authentication, the NTLM algorithm is used to encrypt passwords. It is the algorithm's successor to LANMAN. NTLM had been accompanied by NTLMv2. NTLMv2 uses the HMAC-MD5 hashing algorithm. The LAN Manager hash was one of the first Windows operating systems to use password hashing algorithms and the only one to be supported before the introduction of NTLM used in Windows 2000, XP, Vista and 7. Some newer operating systems tend to allow the use of LM hashes for backward compatibility. This is however unavailable for Windows Vista and Windows 7 by default.

E.LANMAN

Microsoft LANMAN is the Microsoft LAN Manager hashing algorithmLANMAN was used to store passwords on legacy Windows systems. To build the hash LANMAN used DES algorithms. The problem is that LANMAN is not very reliable in implementing the DES algorithm, so LANMAN is susceptible to brute force attacks. In reality, LANMAN password hashes can be cracked in just a few hours. Microsoft no longer uses LANMAN as the default method for storage. It is available but is not turned on by default.

1) Working for Hashing Algorithm

Whereas encoding is essential for data security, it is also necessary to be able to verify that no one has altered the data. Hashing algorithms can achieve so.A hashing is an oneway feature that transforms data in such a way that it is computationally infeasible to generate the original Document, provided a hash result (sometimes called a digest. In addition to being a one-way, hash feature:

- They take an input of any length and produce an output of a fixed length.
- They should be efficient and fast to compute.
- They should be computationally infeasible to invert.
- They should be strongly collision-free.

IV. SHA256 HASHING ALGORITHM

SHA 256 algorithm (sometimes called digest) is a kind of signature for a text or a data file. SHA-256 generates an almost-unique 256-bit (32-byte) signature for a text.

A hash is not' encryption'–it can not be decrypted back to the original text (it is a cryptographic' one-way' feature, and is a fixed size for any source text size). This makes it ideal when comparing' hashed' versions of texts, rather than decrypting the text to obtain the original version.

Such applications include hash tables, integrity verification,

- Challenge handshake authentication
- Anti Tamper
- Digital signatures

A. Challenge Handshake Authentication

Challenge handshake authentication (challenge hash authentication) avoids password transmission in a simple way that a client can send the hash of a password over the internet for validation by a server without the possibility of interception of the original password.

B.Anti Tamper

Anti-tamper –connect a hash of a message to the original, and the receiver will re-hash the Message and compare it to the hash provided: if they match, the Message remains unchanged; this can also be used to confirm no data loss in transmission.

C.Digital signatures

Digital signatures are much more complex, but in essence, by encrypting it with your private key, you can sign a document's hash and create a digital signature for the document. Anyone else will then verify whether you have authenticated the code by decrypting the signature using your public key to recover the original hash and compare it to your code hash.

1) Secure Hash Algorithm Message

Digest Length = 256

Initial hash value:

H[0] = 6A09E667 H[1] = BB67AE85 H[2] = 3C6EF372 H[3] = A54FF53A H[4] = 510E527F H[5] = 9B05688C H[6] = 1F83D9AB H[7] = 5BE0CD19.

Block Contents:

W[0] = 61626380 W[1] = 00000000 W[2] = 00000000 W[3] = 00000000 W[4] = 00000000 W[5] = 00000000 W[6] = 00000000 W[7] = 00000000 W[8] = 00000000 W[9] = 00000000 W[10] = 00000000 W[11] = 00000000 W[12] = 00000000 W[13] = 00000000 W[14] = 00000000 W[15] = 00000018

2) SHA 256 Algorithms Steps

Basic Initialization will be done for 8 items

Step 1: Information is a array 8 things in length where every thing is 32 bits.

Step 2: out is a array 8 things in length where every thing is 32 bit.

Step: 3 Compute all the capacity boxes and store those qualities.

Allude to them by work name

Step: 4 Store input, right moved by 32 bits, into out.

Now, in the out exhibit, E is an inappropriate worth and A is unfilled

Step: 5 Store the capacity boxes.

Presently we have to compute out E and out A.

note: Supplant the modulo orders with a bitwise AND 2^(32-1)

Step : 6 Store (Input I + CH + ((XT+YT) AND 2^31)) AND 2^31 As Mod1

Step : 7 Store (Sum1 + Mod1) AND 2^31 as Mod2

Step : 8 Store (b + Mod2) AND 2^31 into out E

Presently out E is right and all we need is out A

Step : 9 Store (NA + Mod2) AND 2^31 as Mod3

Step : 10 Store (Sum0 + Mod3) AND 2^31 into output A

3) EXCLUSION

```
def SHA256 CE (XT, YT, I, J, K, L, M, N, O, P):

SHA256 Compression Function

CH = (M & N) ^ (~M & O)

NA = (I & J) ^ (I & K) ^ (J & K) #Major

S0 = SS(I, 2) ^ SS(I, 13) ^ SS(I, 22) #Sigma_0

S1 = SS(M, 6) ^ SS(M, 11) ^ SS(M, 25) #Sigma_1

T1 = P + S1 + CH + XT + YT

return (T1 + S0 + NA) & NN, I, J, K, (L + T1) & NN, M, N, O

def SHA256(Z)

Performs SHA256 on an info string

Z: The string to function

return: A 32 byte exhibit of the parallel summary

Z = Pad(Z) # Cushion message with the goal that length is detachable by 64

Dg = list(A) #Digest as 8 32-bit word (I-P)

for j in range(0, len(Z), 64): # Repeat over message in lumps of 64

X = Z[j:j + 64]

Y = [0] * 64

Y[0:16] = [int.from_bytes(X[i:i + 4], 'big')for I in extend (0, 64, 4)]

for i in extend(16, 64):

S0 = SS(Y[i - 15], 7) ^ SS(Y[i - 15], 18) ^ (Y[i - 15] >> 3)

S1 = SS(Y[i - 2], 17) ^ SS(Y[i - 2], 19) ^ (Y[i - 2] >> 10)

Y[i] = (Y[i - 16] + S0 + Y[i-7] + S1) & NN

I, J, K, L, M, N, O, P = DG # Condition of the pressure work

for i in range(64):

I, J, K, L, M, N, O, P = SHA256 CE(Y[i], K[i], I, J, K, L, M, N, O, P)

Dg = [(Q + R) & NN for Q, R in zip(Dg, (I, J, K, L, M, N, O, P))]

return b''.join(Di.to_bytes(4, 'big') for Di in DG)

if _Name_ == "_class_":

qo = SHA256('HI USERS')

print(''.join('{:02x}'.format(i) for i in qo))
```

4) Key Value Stores

The Key-Value Store is constructed as shown in Figure 2 below. The Key-Value Store is a NoSQL-type database that can not manipulate specific data than the SQL database, but has the benefit of fast navigation speeds to access data. The key value is hashed in the table using the hash table to store the keyValue pair. The hashed key values serve as the database location and will be able to access the value. In addition, the Key-Value Store uses an auxiliary hash to avoid collision with a hash function [1].

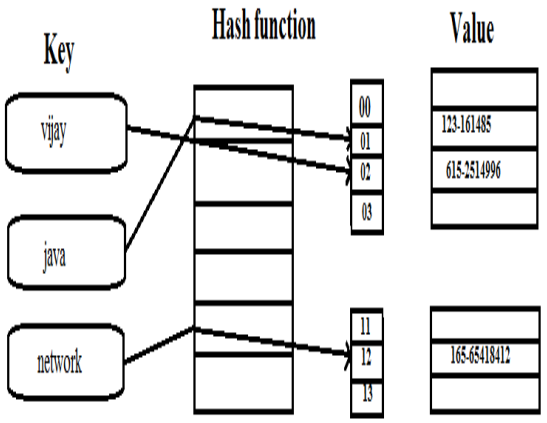

Figure 2. Key-Value Database Using Hash Function

V. A SOCIAL NETWORKING USING BLOCK CHAIN'S ARCHITECTURE

This chapter describes the components of the Social networking System using the Block Chain. The system describes the functions of the Director, Node, Agreement Algorithm, and Smart Contact.

A. Block Format

Frames consist of headers, metadata, and block payloads in the block chain of proposed systems. The block headers include block number, previous blockhead hash, previous transaction hash and previous database hash. The payload contains a list of the transactions carried out with hash. Metadata includes basic block details and the time to create blocks [1].

B. Architecture

The architecture established enviroment of social networking. More than user and one admin user use this social network. SQL Link is an Application Program Interface (API) that allows Synergy applications to access and manipulate data from various database systems using SQL based functions.

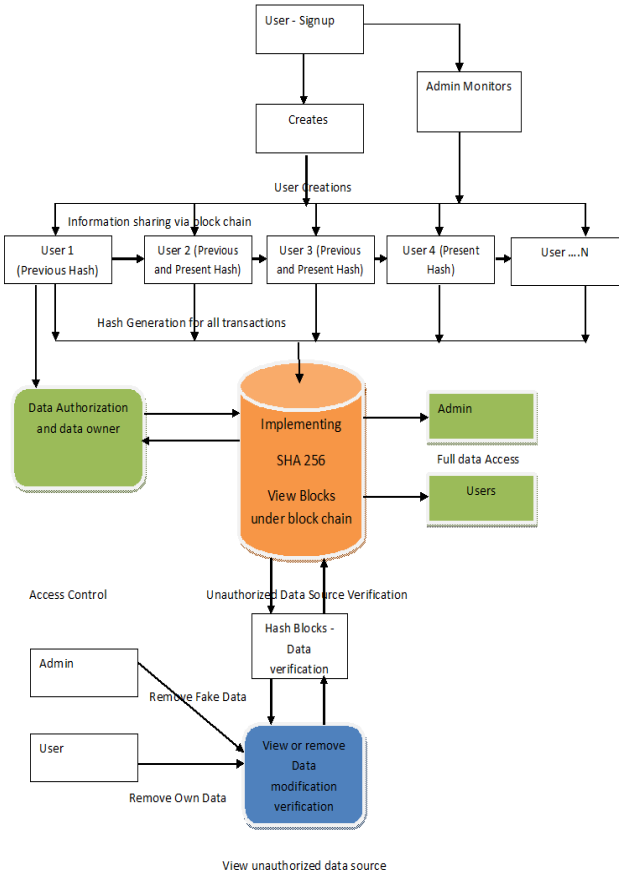

Figure 3. Architecture of Social Network

C. Structure of database

The database consists of a Key-Value store from a database and a hash tree from a database. The Key-Value store is the key value of the hashed identifier of the user and the key value offers access to the value of personal information of the user.

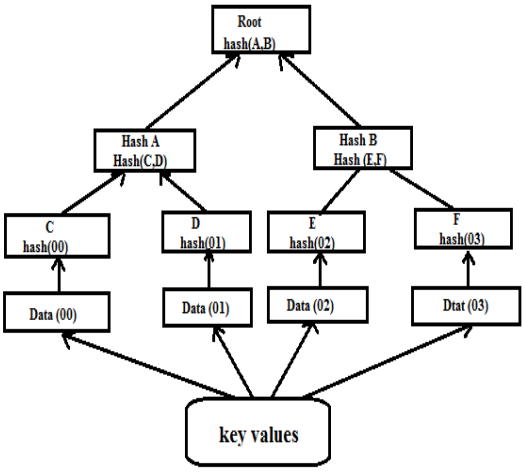

Figure 4. Hash Tree Using Key-Value Store

The database contains a foundation hatchet. Every node in a hash is a value that harms any pair of key-

values. When creating, removing, and modifying data in a database, a hash is created for the corresponding data, and a new one is generated. Even if the database changes regularly, it can build a new one quickly using the hatchet features.

VI. RESULT

The network is more security and any user use the network. This paper major working is Stop spreading fake and old messages around social network and Identifying fake and irreverent Message or news spreaders will be successfully. The Message stop the owner. Any other user not stop the Message. Only message owner stop.

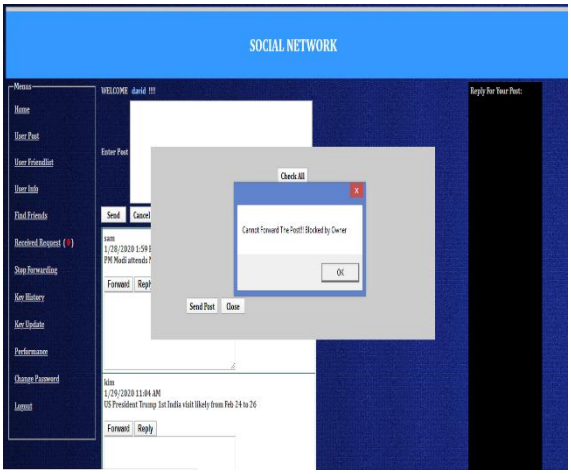

Figure 5. Owner Stopping Message

VII. CONCLUSIONS

Therefore this mission was effectively conducted, and the results was checked.All collected outputs are in abstract as resolved. Initially further issues emerged during the development of the architecture. Architecture was successfully applied as described above. Through engineering both networks function well. And such networks would often operate in a different method. Such apps would render the project more effective and productive.The blockchain creates an internal data transfer for efficient data retrieval from the Social networks.

Displaying results will be more relevant. Now the 89 % of the internet users may get use of this application. This makes more comfortable in the social network application. This makes all the network users are raised to 89 % among the internet users. These features make this project more successful. Harassment users will be thrown out of the network.

REFERENCES

[1] Yongseeon ji, suhwan bae, yongtae shin, "A Personal Information Management using Block Chain", 2019.

[2] Diego Ongaro, John Ousterhout, "In Search of an Understandable Consensus Algorithm(Extended Version)",2014.

[3] Satoshi Nakamoto, Bitcoin:A Peer-to-Peer Electronic Cash System",2009.

[4] Vitalik Buterin. "Ehtereum White Paper A Nect Gerneration Smart Contract & Decentralized Application Platform". 2014.

[5] Muguel Castro, Barbara Liskov, "Practical Byzanine Fault Tolerance", 1999.

[6] Dongyan Huang, Xiaoli Ma, Fellow, "Performance Analysis of the Raft Consencsus Algorithm for Private Blockchains", 2018.