Dynamic Cryptography Integrated Secured Decentralized Applications with Blockchain Programming

https://doi.org/10.31185/wjcm.Vol1.Iss2.41

Jamal Kh-Madhloom (✉)  
Computer Sciences and information Technology College, Wasit University, Iraq

Abstract—Abstract Blocks and chains are the building blocks of the blockchain, which is a community network. Blocks and chains are two terms used to describe collections of data information. The most fundamental need for a blockchain is that these postings be connected by cryptography, which is the case here. Cryptography, the entries in each block are added to as the list grows. Although the concept of blockchain cryptography is difficult, we have made it easier for you to understand. Asymmetric-key cryptography and hash functions are used in blockchains. Hash functions provide participants with a complete image of the internet. The SHA-256 hashing algorithm is often used in blockchains. In Bitcoin, where addresses are tracked by public-private key pairs, blockchains are often used. The public key in blockchain cryptography is a person’s address. All participants have access to the participant’s public key. The private key is used to get access to the address database and to authorise activities using the address. To ensure the integrity of the blockchain ledger, encryption plays a key role. Each event on the blockchain is recorded using encrypted data. As long as each user has access to their cryptographic keys, they may buy or trade cryptocurrencies. The root hashes of all transactions are stored in blockchains via cryptographic hashing. If somebody attempts to tamper with any data upon that blockchain, the main hash will have a completely new hash. Root hash comparisons may be performed on any other system to check whether the data is safe.

Keywords—Blockchain, Dynamic Cryptography, Security in Blockchain

1 Introduction

As a leading technology, blockchain is usually linked with high levels of security and anonymity across a wide range of applications. Blockchain technology is now being used in a wide range of social and business contexts, not only in the cryptocurrency industry. E-governance, social networking and e-commerce are just a few of the many sub-segments that fall under this umbrella [1].
Secured Digital Ledgers with Cryptography in Blockchain

Digital ledgers are stored on a high-security, high-performance system known as a "blockchain." Using a digital ledger eliminates the need for intermediaries or administrators to manipulate the records [2]. All operations on the bitcoin blockchain are finalised using a variety of protocols and processes that cannot be hacked by external parties. [3].

Key Implementations of Blockchain Technology

- Entertainment
  - Spotify
  - Guts
  - B2Expand
  - KickCity
  - Vereditum

- Social Networks
  - Matchpool
  - MeWe
  - Minds
  - Steepshot
  - Mastodon
  - DTube
  - Sola

- Retail
  - Opskins
  - Loyyal
  - Warranteer
  - Every.Shop
  - Blockpoint
  - Fluz Fluz
  - Spl.yt
  - Shopin
  - Ecoinmerce.io
  - Portion
  - Buying.com

- Cryptocurrency
  - Litecoin
  - Ripple
Primecoin  
Bitcoin  
Namecoin  
Dogecoin  
Nxt  
Ethereum

2 Real Problem and Key Statement

In traditional centralized environment, there is less security because of weak hashing and needs dynamic hash with cryptography as in Blockchain. Transactions cannot be reversed because cryptographic hashing is irreversible. This assures that all users can rely on the digital ledger's correctness and that they are protected from any antagonistic conduct [4, 5].

To understand blockchain, one must understand cryptography. It is possible to encrypt, transmit bitcoin securely, and record transactions over time thanks to cryptography's features. Without a central authority, we may trade bitcoin safely and assure that blocks will continue to be inserted into the chain without restriction [6].

Cryptographic hashing enables blockchains to store vast quantities of transactions while protecting them from hackers. It provides a secure, verifiable, and scalable method for conducting online transactions. Blockchain is genuinely unstoppable because of cryptography [7, 8].

2.1 Public and Private Blockchain with Security and Cryptography

The data structure of blockchain technology is automatically safe. Encryption and decentralisation are the foundations upon which it is based to ensure transactional trust. In most blockchains and DLT, each block of data contains a single transaction or a collection of related transactions. It is extremely impossible to tamper with a cryptographic chain since each new block is irrevocably tied to the blocks that came before it. To ensure that each transaction contained in a block is correct, each transaction must be verified and agreed upon by the consensus process [9, 10].

Blockchain technology is able to transform society since it relies on the participation of all users in a network [11, 12]. There is no single point of failure because everything is documented. But when it comes to digital money security, blockchain technology has a few advantages over the alternatives. Participation and data access might differ amongst blockchain networks. Public and private networks are the two most common varieties.
A common feature of public blockchain networks is the ability for anybody to join while yet maintaining their anonymity. Using computers linked to the internet, a public blockchain can verify transactions [13] and establish consensus. Public blockchains, like as Bitcoin, use "bitcoin mining" as a means of bringing parties together. The "miners" on the bitcoin network work cooperatively to solve a complex cryptographic challenge in order to validate a transaction. Other than public keys, this type of network has few means of identity and access control.

In private blockchains, where only well-known companies are allowed to participate, identity is employed to authenticate membership and access credentials [14, 15, 16]. The organisations' combined efforts establish an exclusive "business network." Consensus on a private blockchain in a permissioned network is achieved by "selective endorsement."

Before choosing a network for any blockchain application, think about the requirements of your firm. It is better for requirements to use permissioned and private networks, as they may be more tightly controlled [17, 18]. Decentralization and dispersion can take place more easily in permissionless networks. Table 1 is the key tools for secured blockchain

<table>
<thead>
<tr>
<th>URL</th>
<th>Framework / Tool</th>
</tr>
</thead>
</table>

Table 1. Key Tools for Secured Blockchain Environment
3 Key Objective and Related Aspects

The unique verification process of Blockchain is one of its outstanding features. By removing the need for human verification, Blockchain promises to increase accuracy. Third-party verification costs have also been reduced as a result of the use of the blockchain technology [19]. Because of its decentralised design, hackers and attackers have had a tough time tampering with data. Security, privacy, and efficiency are just a few of the benefits of using Blockchain to do business. Since it is a transparent technology, it provides complete transparency to its users. As a result, inhabitants of countries with unsafe or undeveloped governments can use Blockchain as a financial option as well as a way to protect their personal information [20]. It's still early days for blockchain, but there are countless chances for professionals to study and grow their careers in this industry, including cryptography in blockchain for sure.

There are blocks of various data items and documents in the blockchain network. It is impossible to alter a block after it has been added to the blockchain. In this context, “immutable” refers to the fact that it cannot be changed or tampered with. Consequently, it creates a secure chain of blocks that eliminates the risk of data manipulation or leaking [21–24].

The Genesis Block is the initial entry in the chain of the network and is where the blockchain begins transactions. It gets more difficult to decipher the prior states because of the various encryptions that have been added to each new block that is inserted. Table 2 show the literature review.

Table 2. Literature Review

<table>
<thead>
<tr>
<th>Authors</th>
<th>Key Work</th>
</tr>
</thead>
<tbody>
<tr>
<td>G. Zyskind, A.S. Pentland [25]</td>
<td>Using Secured Cryptography based blockchain, data exchanges between users and apps may be protected and undamaged. Network nodes reward trustworthy nodes for their degree of trust rather than proof-of-work.</td>
</tr>
</tbody>
</table>

A. Ouaddah, A. Abou Elkalam, A. Ait Ouahman [27] PoC Based Implementation

M. Ali, et al. [28] Dynamic Cryptography based application., “BlockStack” is a test project for immutable data naming and storage. Recognizing that the Namecoin blockchain does not provide the same level of security and trustworthiness as the Bitcoin blockchain.

K. Christidis, M. Devetsikiotis [29] IoT devices with implementation patterns with the use of blockchain technology are examined in detail.

4 Methodology

- Using Python based Programming Platform for Cryptography in Blockchain Development
- Development, Generation and Deployment of Dynamic Hash
- Using Dynamic Hash in Blockchain Environment

Python is the programming language of choice for high-performance computations in practically every field. The tools and frameworks provided by Python may be used to construct blockchain applications, including those that are decentralized.

$ pip install <packagename>

MyDrive:\PythonInstallationDirectory>python -m pip install <packagename>

The incorporation of dynamic cryptographic and encryption into the blockchain technology is critical and heavily relied upon. Installing the hashlib library is as simple as following the instructions found in the previous paragraph.

To ensure that all transactions and records are safe, a secure blockchain generates hash values for each transaction and record. It is possible to produce the dynamic hash value needed to build a blockchain from a collection of individual transactions by running the following script, blockchainhash.py.

```python
import datetime as date
import hashlib as hashlib

class Declaration BlockchainId:
```
```python
    def __init__(self, myindex, ts, myBlockchainIdchain, backhashlibrary):
        self.myindex = myindex
        self.ts = ts
        self.myBlockchainIdchain = myBlockchainIdchain
        self.backhashlibrary = backhashlibrary
        self.hashlibrary = self.hashlibraryop()

    def hashlibraryop(self):
        shahashlibrary = hashlibraryer.sha256()
        shahashlibrary.update(str(self.myindex) + str(self.ts) + str(self.myBlockchainIdchain) + str(self.backhashlibrary))
        return shahashlibraryhexdigest()

def initializeBlockchainId():
    return BlockchainId(0, date.datetime.now(), "InitializeBlockchainId BlockchainId", "0")

def next_BlockchainId(last_BlockchainId):
    this_myindex = last_BlockchainId.myindex + 1
    this_ts = date.datetime.now()
    this_myBlockchainIdchain = "BlockchainId" + str(this_myindex)
    this_hashlibrary = last_BlockchainId.hashlibrary
    return BlockchainId(this_myindex, this_ts, this_myBlockchainIdchain, this_hashlibrary)

BlockchainIdchain = [initializeBlockchainId()]
back_BlockchainId = BlockchainIdchain[0]
maxBlockchainIds = 20
for i in range(0, maxBlockchainIds):
    BlockchainId_to_add = next_BlockchainId(back_BlockchainId)
    BlockchainIdchain.append(BlockchainId_to_add)
    back_BlockchainId = BlockchainId_to_add
    print "BlockchainId #{} inserted in BlockchainId-chain".format(BlockchainId_to_add.myindex)
    print "Hashlibrary Value: {}
    print "{}\n".format(BlockchainId_to_add.hashlibrary)
```
5 Result

The result of executing code is a different hash value and a better degree of security employing cryptography techniques. Attempts to hack or sniff the transaction will be nearly impossible if these hash values are used. Figure 2 shows the result of secured hash generation in blockchain.

![Secured Hash Generation in Blockchain](image)

6 Deployment of Network Based Distributed Blockchains

Block-based hash functions, like those in preceding examples, are implemented on a standalone system. The real blockchain must be dispersed in order for various users to start their own transactions and blocks. Python has a variety of frameworks for distributed and web-based solutions.

Thus, the digital currency or transaction is carried out securely. B's records need to reflect the value of a file or digital currency sent by A, for example, if A's records need to be wiped and mirrored in B's records. Traditionally, the Bank has acted as a go-between in this transaction. On the blockchain, transactions are validated in real time by specialised algorithms, cutting out the middlemen. Currency will depreciate regardless of the kind of currency if the sender does not erase the transaction from their account. Fig. 3. Shows blockchain initialization.

```python
CryptoMiner_address = "***************************"
mySecuredBlockchain = []
mySecuredBlockchain.append(Creation_genesis_SecuredBlock())
ThisClass_Secured_Nodes_SecuredTransactions = []
peer_Secured_Nodes = []
SecuredMining = True
```
@Secured_Node.route('/mySecuredBlockchain', methods=['POST'])
def SecuredTransaction():
    new_mySecuredBlockchain = request.get_json()
    Class_Secured_Nodes_SecuredTransactions.append(new_mySecuredBlockchain)
    Display "New SecuredTransaction"
    Display "Sender: {}".format(new_mySecuredBlockchain['from'].encode('ascii', 'replace'))
    Display "Receiver: {}".format(new_mySecuredBlockchain['to'].encode('ascii', 'replace'))
    Display "Amount: {}n".format(new_mySecuredBlockchain['amount'])
    return "SecuredTransaction Successful"

@Secured_Node.route('/SecuredBlocks', methods=['GET'])
def get_SecuredBlocks():
    chain_to_send = mySecuredBlockchain
    for i in range(len(chain_to_send)):
        SecuredBlock = chain_to_send[i]
        SecuredBlock_idx = str(SecuredBlock.idx)
        SecuredBlock_timestamp = str(SecuredBlock.timestamp)
        SecuredBlock_data = str(SecuredBlock.data)
        SecuredBlock_hash = SecuredBlock.hash
        chain_to_send[i] = {
            "idx": SecuredBlock_idx,
            "timestamp": SecuredBlock_timestamp,
            "data": SecuredBlock_data,
            "hash": SecuredBlock_hash
        }

Fig. 3. Blockchain Initialization
$ curl "http://localhost:5000/blockchain" -d "{"from":"ss","to":"fsd", "amount":3}" -H "Content-Type:application/json"

Fig. 4. Curl based Cryptography Hash Initialization

Proof-of-work (PoW) is a critical algorithm in blockchain programming. It is used to verify and confirm transactions in order to add new blocks to the blockchain. The key consensus algorithm for verifying and authenticating transactions is referred to as such. In a blockchain network, there are a variety of miners that work together to verify and finalise transactions. The miners are compensated with digital cryptocurrencies as compensation for their successful validations. Figure 4-6 shows the execution process.

Fig. 5. Secured Transactions
With the code execution and overall implementation described, there will be no effort at hacking because all data and transactions can be inspected to ensure complete transparency. It is possible to record and commit the integrity of transactions by utilising Proof of Work (PoW).

7 Conclusion

Currently, governments and corporations alike are working to protect their applications by implementing blockchain technology. Secure Proof of Work (PoW) algorithms must be linked to these integrations in order to ensure implementation privacy and integrity. Blockchain technology may be utilised by researchers and forensic scientists to accurately anticipate the identities of certain individuals, which can be employed in criminal forensic and law enforcement settings. Software (Electrum, Bitcoin core) and perhaps a specific hardware device (e.g. Ledger) can be used to store transaction data and the user's private and public keys (e.g. private/public key pair). It's critical to understand that these wallets do not hold any actual money (e.g. Bitcoin, Ethereum). There is nothing more to these wallets than a location to store one's private keys and transaction balance. A blockchain wallet is also required to conduct transactions with these other users. To put it another way, the blockchain holds all of the true information/data and cash in blocks, not a wallet. It's akin to a digital signature, which serves as a kind of identification for both the recipient and the whole blockchain network. A particular method must be used to combine your data and your cryptographic signature to establish a unique digital signature each time you begin a transaction with another node. As a result, you may rest assured that both your node and also the data it transmits are genuine.
8 References


[29] K. Christidis, M. Devetsikiotis, Blockchains and Smart Contracts for the Internet of Things, IEEE Access

Article submitted 3 July 2022. Published as resubmitted by the authors 1 August 2022.