IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 09 Issue 07, July 2022
ISSN (Online) 2348 – 7968 | Impact Factor – 6.72
www.ijiset.com

A Blockchain-Based Shamir's Threshold Cryptography Scheme for Data Protection with Overlapping Data Distribution

Muhsina K ,

(Applied Electronics And Communication Systems), MGM College Of Engineering and Pharmaceutical Sciences ,Valanchery, India, muhsivalliyengal@gmail.com

Padmam Gopinath Kaimal, Asst Professor
,(Applied Electronics And Communication Systems) ,MGM College Of Engineering and Pharmaceutical Sciences ,Valanchery, India, padmamkaimal@gmail.com

Abstract: Blockchain is a growing list of linked blocks. It is a paradigm-shifting technology that has emerged over the past decade. Which is based on peer-to-peer communication technology, network theory, and cryptography. This is a cryptographic method for data protection with overlapping data distribution method. Encryption is commonly used to ensure privacy and confidentiality of the IIoT data. Commonly the encrypted data is stored in the cloud and the keys are directly stored, their exist security and privacy risk. These problems are overcome by Shamir's Threshold Cryptography approach that utilizes Blockchain (STCChain).In this paper proposed a Shamir threshold cryptography scheme for data protection using blockchain with Overlapping Data Distribution. Blockchain stores the whole list of transactions, so that the number of transaction increases storage space also increases .So that changing the storage concept by using Overlapping Data Distribution. STCChain solves the security and privacy issues of key also overlapping data distribution method reduce the overlapping data distribution storage requirement..

Index Terms—blockchain, cryptography, overlapping data distribution, IIoT, Shamir's Threshold Cryptography, STCChain.

1 .INTRODUCTION:

Commerce on the Internet has come to rely relatively exclusively on financial institutions serving as trusted third parties to process electronic payments. While the system works well enough for most transactions, it still suffers from the inherent weaknesses of the trust based model . The blockchain technique, which is a distributed solution of the trust problem without any third party, is a promising substitution . Blockchain as the name indicates, a chain of blocks that contains information. The blocks consist of valid transactions, a timestamp and a hash pointer as a link to the previous block in the chain. Each transaction in the public ledger need to be verified by participants with consensual majority. Blockchain is a distributed database of records or public ledger of all time stamped transactions saved in all computers in one peer-to-peer network. This is a growing list of linked blocks.

Data sensed, collected and disseminated by the IIoT devices are generally stored and processed in the cloud. To minimize the risk of data leakage , protective measures such as data encryption are used . Such approaches generally require the data encryption/decryption key to be stored and/or managed directly by users or by a centralized third-party institution. There are some limitations exist. In this project use the Shamir's Threshold Cryptography in blockchain (STCChain) with overlapping data distribution. The key is encrypted then dividing into parts by using Shamir's Threshold Cryptography. The parts are stored in

blockchain . The blockchain stores the whole list of transactions. As the number of transaction increases blockchain size also increases. To reduce the storage requirement use the overlapping data distribution. It is a method to divide the whole blockchain of transactions into some non-overlapping shards and make multiple copies of them. Then, distribute these shards uniformly across the nodes in the network. STCChain can effectively prevent attackers from stealing data as well as ensuring the security of the encryption key.

2 .RELATED WORKS

Literature survey is done for specified papers which are essential to know the existing techniques their significance and limitations. It also includes various supporting papers for the proposed technique and their advantages .The Industrial Internet of Things is a system composed of networked smart objects, network physical assets, related general information technology, and, in some settings, cloud or edge computing platforms that support various processes and services in an industrial environment .In this section, we briefly review several related studies on IIoT data encryption and existing blockchainbased approaches and Methods for reducing storage in blockchain . In [1],A. Bahga et al. proposed Internet of Things (IoT) are being adopted for industrial and manufacturing applications such as manufacturing automation, remote machine diagnostics, health management of industrial machines and supply chain management. Cloud Based Manufacturing is a recent on-demand model of manufacturing in IoT technologies. In this discussed a decentralized, peer-to-peer platform called BPIIoT for Industrial Internet of Things based on the Block chain technology. The BPIIoT platform enables a marketplace of manufacturing services where the machines have their own Blockchain accounts and the users are able to provision and transact with the machines directly to avail manufacturing services. In [2],L. Tan et al.proposed IIoT is a typical infrastructure of a smart city. Due to the limited storage and computing capabilities of industrial IoT devices, the data sensed and collected by these devices is usually stored in the cloud, and encryption is usually used to ensure the privacy and confidentiality of sensitive data in the cloud. However, since the keys used for data encryption and decryption are usually difficult to manage, it is easy to cause security and privacy issues. To this end, this is a blockchain-based IIoT data protection scheme. In this scheme prevent the loss of the private key and the disclosure of privacy, use the Shamir secret sharing algorithm to split and encrypt the private key, and publish it on the blockchain.

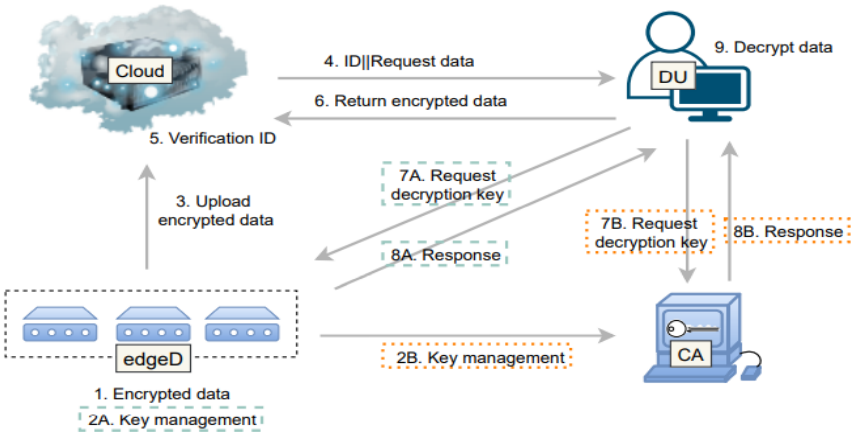

Figure 2.1: Current IIoT data storage model.

The above figure 2.1 is current IIoT data storage model. The limitations associated with the local storage of keys include single point of failure or attack and loss of data or service access if the local storage medium is corrupted (unless there is a backup copy of the key on another storage medium). If the keys are stored and managed by a trusted third-party center (e.g., certificate management organization (CA)), then we must trust the trusted third-party center to do the right thing and not leak the key. In [3], N. Kumar et al.proposed IIoT data in the cloud server achieve cost saving and collaboration .This is a secure channel free certificateless searchable public key encryption with multiple keywords (SCF-MCLPEKS) scheme for IIoT deployment. Then demonstrate the security of the scheme in the random oracle model against two types of adversaries, where one adversary is given the power to choose a random public key instead of the user's public key and another adversary is allowed to learn the system master key. In presence of these types of adversaries, evaluated the performance of this scheme and demonstrate that it achieves (computational) efficiency with low communication cost. In [4], Y. Zhang et al.proposed a privacy-preserving and pairing-free multi-recipient CLKS scheme for cloud-assisted IIoTs. This scheme has the following merits: (1) supporting multi-recipient keyword search function; (2) requiring no costly bilinear pairing operations; (3) providing resistance against keyword guessing attacks. The performance 3 comparison and analysis demonstrate that it is more efficient than the existing CLKS schemes and is appropriate for the cloud-assisted IIoTs.

In [5],M. Dai et al. proposed a low-storage Blockchain(BC) system that employs network coding theory to divide the transaction data into multiple blocks and then stores the blocks at different nodes. This can realize distributed storage (DS) by recovering the transaction data through network coding (NC).This scheme have two kinds of NCbased DS, one is deterministic rate (NC-DRDS) and the other is rateless (NC-RLDS), to deal with a fixed and variable number of blockchain nodes.In [6],A. Dorri et al. proposed Memory Optimized and Flexible BC (MOF-BC)allows the user to summarize or remove part of the "aged" BC transactions. This eventually reduces the ability to perform a public audit, as some of the information is eventually erased.MOF-BC that empowers users and Service Providers (SPs) to remove a previously stored transaction or reduce its size by summarizing transactions or aging the data in transactions. The user/SP may decide to offload the associated overheads for optimizing BC memory to the network using Network-Initiated Memory Optimization (NIMO). To encourage users/SPs to employ memory optimization, MOF-BC offers flexible transaction fees and rewards. In [7] , R.K .Raman is first introduced the idea of distributed storage blockchain (DSB) .The blockchain transaction block is first encrypted and then stored at different nodes (together with the encryption key) using a secret-sharing scheme. In this way, the storage of a blockchain is reduced significantly In [8],D. Mechkaros et al.proposed a secret-sharing scheme is proposed to reduce the size of the blockchain transactions. Each transaction block is divided into t parts, and the size of each part is 1/t size of transaction block.Use the secret-sharing mechanism to share t parts into n shares. Hence, each node stores not one transaction but one share in the blockchain system. Also reduce the storage cost of a blockchain transaction by 1/t without introducing an additional recovery communication cost.This scheme is more efficient and secure compared to other existing schemes that aim to reduce BC storage for industrial big data.

3. BLOCKCHAIN TECHNOLOGY

3.1 BLOCKCHAIN

Blockchain is a paradigm-shifting technology that has emerged over the past decade, which is based on peer-to-peer communication technology, network theory, and cryptography . In a traditional centralized database model, the central server stores every transaction record, which can be disastrous if the central server is being attacked. In contrast, blockchain technology is a distributed solution without any third-party trust problem.Since every node in the blockchain network keeps a copy of the public ledger, it is possible to audit the transactions locally without referring to a centralized authority .Blockchain diagram[9] shown in Figure 3.1.

Figure 3.1: Blockchain

Therefore, blockchain can also be viewed as a distributed ledger system, which eliminates the disadvantage of a traditional centralized database model .However, the storage cost in blockchain is huge , which is one of the factors currently limiting the widespread adoption of blockchain technology. Since each node needs to store the public ledger locally, the storage cost of an entire blockchain network grows quadratic ally .Blockchain is an emerging technology that promises many exciting applications in various fields, including financial, medical, energy, and logistics management. However, there are still some limitations in the existing blockchain framework that prevents its widespread adoption in the commercial world. One important limitation is the storage requirement. Therefore ,need to change the storage concept in blockchain.

The transactions are time stamped and bundled into blocks where each block is identified by its cryptographic hash. The blocks form a linear sequence where each block references the hash of the previous block, forming a chain of blocks called the Blockchain. A Blockchain is maintained by a network of nodes and every node executes and records the same transactions. The Blockchain is replicated among the nodes in the Blockchain network. Any node in the network can read the transactions.

3.1.1 How does blockchain works?

The process of completing the transactions in blockchain is presented in Figure 3.2. When someone requests a transaction or two parties exchange data, that can be digitally described; the requested transaction is broadcast to peer-to-peer network consisting of computers called nodes. This network of nodes validates the transaction and the status of the user using known algorithms.

The transaction can be verified immediately or can be transcribed securely in a list of pending transactions. Blocks in the chain are identified by a hash and each block contains a group of transactions and a header which is a reference to the hash of the previous block. This list of linked blocks creates an interdependent and secure chain. A block need to be validated, in order to be added into a blockchain. When a block is verified, it is distributed through the network, i.e., added to the existing chain. Then the transaction is completed, and it is recorded in a public ledger.

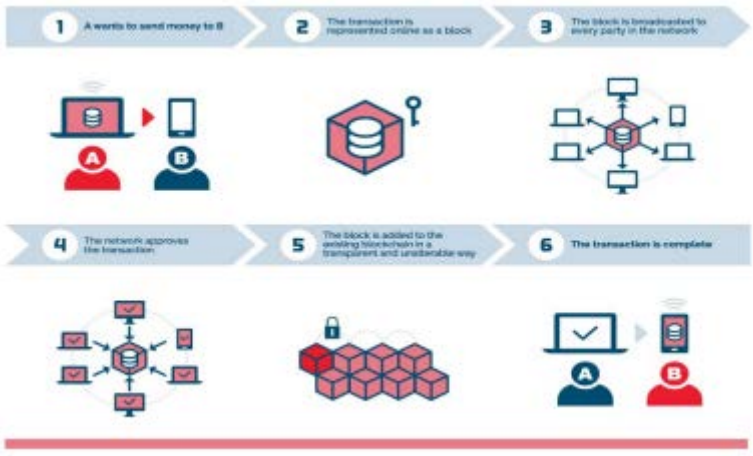

Figure 3.2: Working of a blockchain

3.1.2 Blockchain size

Blockchain is a shared distributed database, which can be agreed upon a peer-to-peer network. It contains a connected sequence of blocks, holding time stamped transactions. The confidence is based on the security provided by the public-key cryptography and verified by the network community. Once a block is appended to the blockchain, it cannot be changed. This feature makes the blockchain unchangeable. As we mention above, each node stores all transactions. This means that as the number of transactions increases the storage space and cost rapidly grows.

In Figure 3.3 the total size of all block headers and transactions in Bitcoin blockchain over the last year is given. Blockchain is a distributed ledger system that requires each participating node to store a copy of ledger for the transaction records. Every time a transaction block is created, it is first verified by the neighbouring nodes and then goes

through the consensus (mining) process. This transaction block is then added to the ledger, where in each block is related to the previous block through a chain of hash values. This data structure suffers from scalability issue, as the storage room required to keep the entire ledger is growing quadratic-ally when the number of blockchain nodes increases.

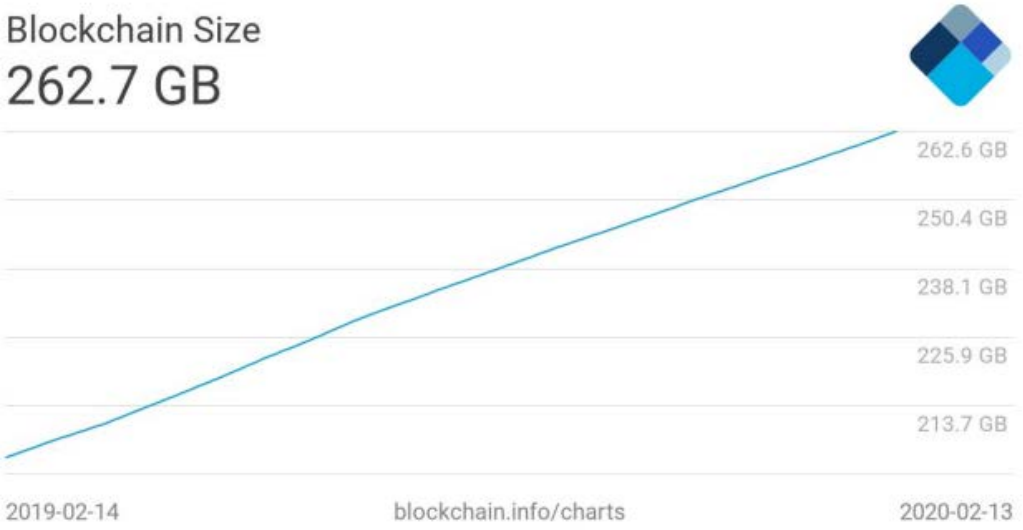

Figure 3.3: Blockchain size

4. SHAMIR'S SECRET SHARING

Adi Shamir introduced the concept of secret sharing through a (k, n) threshold scheme. His model was based on polynomials. The concept of Shamir's secret-sharing scheme is to distribute a secret S to a group of n participants, so that each participant receives one share of the secret. Knowledge of any k or more shares makes S easy to compute, but less than k shares do not reconstruct the secret message S. This scheme has two phases Secret distribution and Secret reconstruction.

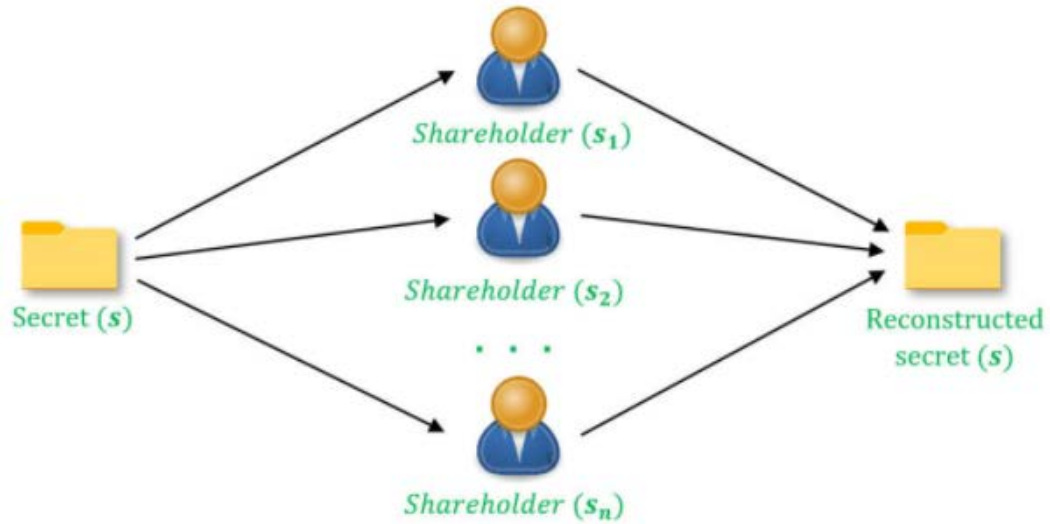

Figure 4.1: Secret Sharing between n participants

IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 09 Issue 07, July 2022
ISSN (Online) 2348 – 7968 | Impact Factor – 6.72
www.ijiset.com

4.1 SECRET DISTRIBUTION

The concept of Shamir's secret-sharing scheme is to distribute a secret S to a group of n participants, so that each participant receives one share of the secret. In order to share the message S, a random (k–1)-degree polynomial he secret message S is the constant term, i.e., Any k or more shares required to recovery

$$f(x)= a_0 + a_1 x + \cdots .. + a_{k-1} x^{k-1} \quad (4.1)$$

$$S=f(0)= a_0 \quad (4.2)$$

coefficients $a_0, a_1, \ldots .. a_{k-1}$ are randomly chosen coefficients from a finite field Fp.where p is prime number; the secret message S is the constant term. The n points (shares) (i, f(i)) for different values of i (for instance i = 1,2,. . . , n) together with the prime p, are given to n participants (nodes) .

Figure 4.2 shows an example of Shamir's (3, 6) secret sharing, which can be adopted by blockchain. In this sharing scheme, each block is stored in a distributed manner among all nodes.

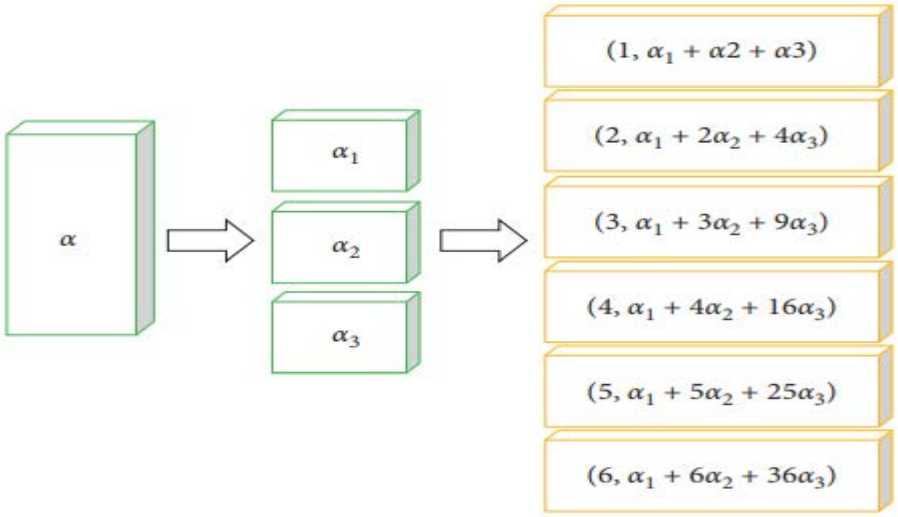

Figure 4.2: Optional (3, 6) Shamir secret sharing

4.2 SECRET RECONSTRUCTION

Any k or more shares are required to recovery of the secret data .For every participant obtains one point (i, f(i)) and with any subset of k of these points, the coefficients of the polynomial can be find using interpolation. Legrange basis polynomials is used to the data recovery process.Knowledge of any k or more shares makes S easy to compute, but less than k shares do not reconstruct the secret message S.. Shamir's (k, n) secret sharing scheme has the

several advantages. Some of them are: 1) For fixed k, we can dynamically add or delete pieces without changing the other pieces; 2) We can easy to improve the security without changing the secret, constructing only new shares for the participants, keeping the same polynomial constant term; 3)If the hierarchy of partici-pants is important, we can give a different number of pieces to participants according to their level in the hierarchy.

5. STCCHAIN WITH OVERLAPPING DATA DISTRIBUTION

5.1 STCChain

Shamir's Threshold Cryptography approach that utilizes blockChain (STCChain)for key data protection.That is STCChain[12] is a combination of blockchain and shamir's secret sharing scheme. STCChain is a new threshold encryption and protection scheme for critical data based on blockchain. The current IIoT data storage model still has the following two problems:

• Problem 1. edgeD storage has limited computing power, and it is difficult to encrypt data autonomously. Generally, the encryption method is composed of a symmetric and an asymmetric key. However, even if symmetric key encryption can provide a lightweight solution for edgeD, due to the low capacity and low performance of edgeD, autonomous encryption is very difficult.

•Problem 2. Centralized key storage is vulnerable to attacks. On the one hand, malicious CA administrators could illegally use edgeD encryption and decryption data keys, thereby stealing edgeD data and causing privacy leakage. On the other hand, an attacker could attack the CA key database and steal edgeD data, thereby destroying data confidentiality and privacy. In addition, edgeD stores the key locally,which is prone to single point of failure and privacy leakage.STCChain provide solution for the two limitations of encryption/decryption key protection mentioned above.

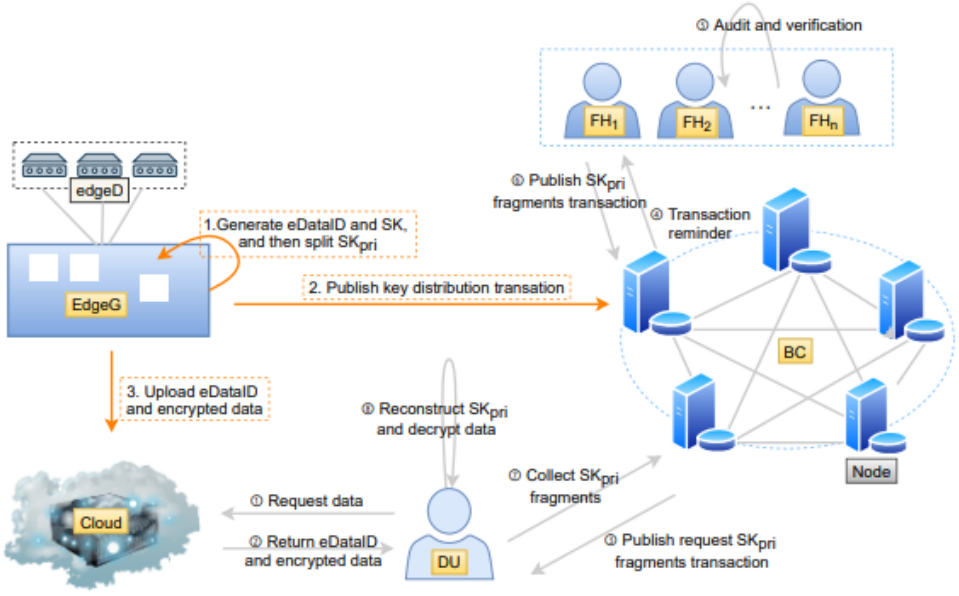

Figure 5.1: STCChain architecture

The STCChain model is shown in Figure 3.1, including the following five entities:
(1)EdgeG: Edge gateway, which is responsible for processing the data uploaded by edgeD.

(2) BC: Blockchain, which is open, transparent, tamper proof, and irreversible. It is the same as the distributed database, and we use it as a key storage database for STCChain. (3) Cloud: Cloud storage, which provides identity authentication for DU and non-real- time data storage for EdgeG. Cloud verifies the identity of the DU and returns eDataID to the user.
(4)DU: Data users, if DU obtains encrypted data from Cloud and collects a sufficient number of key fragments from BC, the data can be decrypted.
(5) F H: If the holder of the key fragment of eDataID verifies the identity of the DU, he or she will send his or her owner key fragment to DU.

The overall process is as follows:
Step 1. edgeD uploads the data to EdgeG. After EdgeG receives the data, EdgeG generates eDataID, uses ksm (the symmetric key generated by EdgeG) to encrypt the data, encrypts ksm by SKpub (public key in the asymmetric key SK generated by EdgeG) and uploads the encrypted data to Cloud. Finally, EdgeG uses the SSS algorithm to split the decryption key SKpri (private key of SK) to obtain n key fragments.
Step 2. A smart contract is used to publish key distribution transactions; that is, each key fragment is encrypted with each F H's public key and then published to BC for storage.
Step 3. Cloud uploads eDataID and the encrypted data to Cloud. If DU wants to access resources, first, DU sends an access request to the Cloud, Cloud verifies DU, and then eDataID is returned to DU. Next, DU initiates a private key transaction to BC, triggering BC to send a transaction reminder to F Hi(i = 1, 2, ..., n). Each FHi verifies and then uses a smart contract to publish the key fragment transaction to BC, where each key fragment uses DU's public key for encryption to ensure privacy. Finally, DU collects a sufficient number of key fragments .

5.2 Overlapping Data Distribution

In this project STCChain uses the overlapping data distribution method[13].Blockchain has a limitation ,that it stores the whole list of transactions .So that the number of transaction increases the storage requirement and blockchain size also increases . Overlapping Data Distribution is a method to divide the whole blockchain of transactions into some non-overlapping shards and make multiple copies of them. Then, distribute these shards uniformly across the nodes in the network.Theoretically this approach not only improves the storage requirement but also ensures the integrity of the data in blockchain in case of node failures. To reduce the amount of storage needed in the network,using overlapping data distribution method.

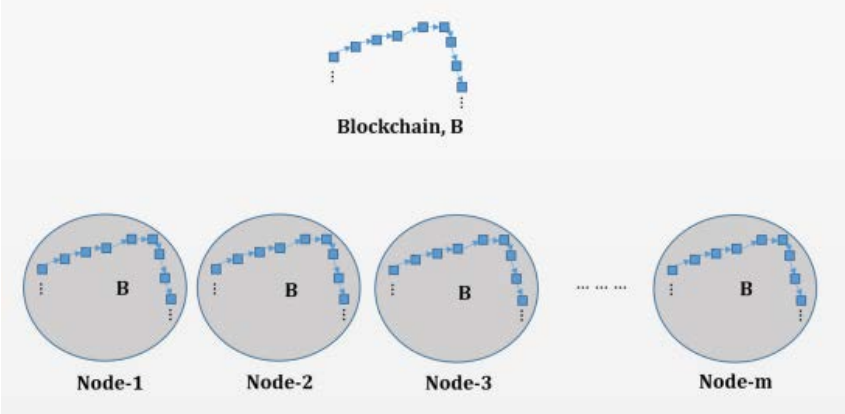

Figure 5.2: Blockchain network before using data sharding

Storage Problem[14] is the main problem in a blockchain network, public or private. In a blockchain network, each node stores one copy of the entire blockchain. If the number of existing blocks is high enough, the required storage to store the whole blockchain can get very large. For example, the Bitcoin network takes up around 242 GB of space as of September 2019 [15] and this number is continually increasing by around 0.1 GB per day. Estimations say that Bitcoin blockchain will take up around 40 TB in 20 years from now. This will put outrageous load on required storage at each node. Things can get equally bad for private blockchains with large number of block. In BC Without using any data sharding, each node in the blockchain network needs to store the entire chain of blocks as displayed in Fig. 3.2 Here, the total storage requirement is $m*B$ when m denotes the number of nodes in the network and B denotes the blockchain size. First, gather information about the number of nodes or make estimation if needed. Then divide the blockchain into some non-overlapping shards after making multiple copies of the blockchain to ensure fault tolerance. Then, put a certain number of shards at each node in the network and thereby reduce the amount of storage needed. In this process, we also make sure to provide fault tolerance by having multiple copies of a shard across the nodes in the network. Mathematical analysis shows

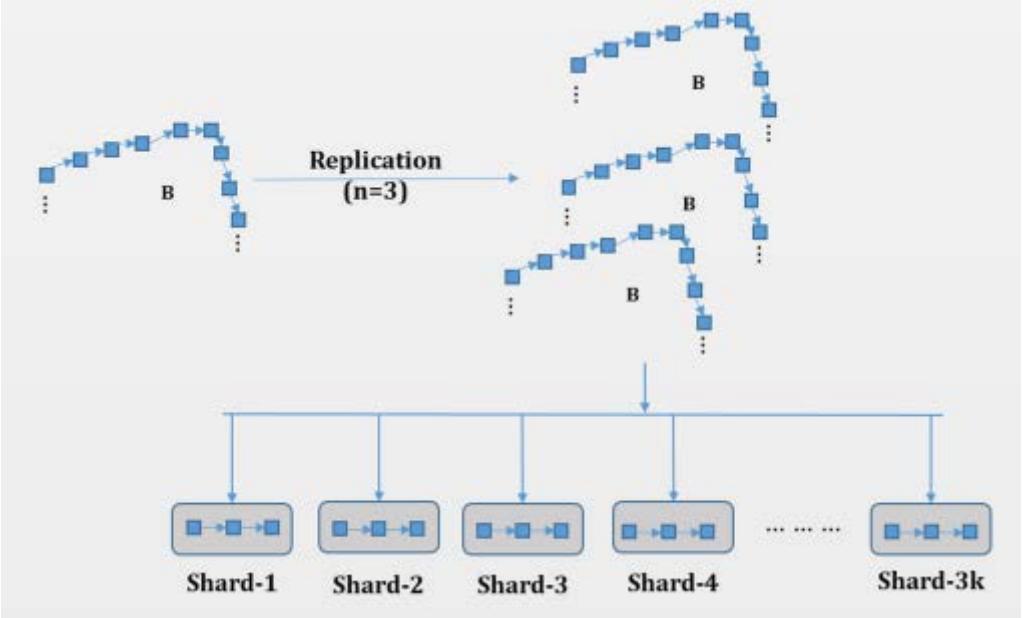

Figure 5.3: Dividing the blockchain into shards

that,this method can not only reduce the storage requirement greatly but also ensure fault tolerance based on the selected parameters. To ensure fault tolerance, we need to make sure that there are multiple copies of a block in the network. Thus, we need to make a number of copies (n) of the initial blockchain first. This number should go up there we expect more fault tolerance from a system. The replication technique for n = 3 is shown in Fig.5.3.

Let us consider that we generate k shards from each copy of a blockchain. This will leave us with a total of n ∗ k shards in the system (3k in the example). Now, we have to distribute these n ∗ k shards across m nodes. Let us consider that we put p shards at each of the m nodes to use up all the n ∗ k shards. Figure 5.4 shows the distribution process.

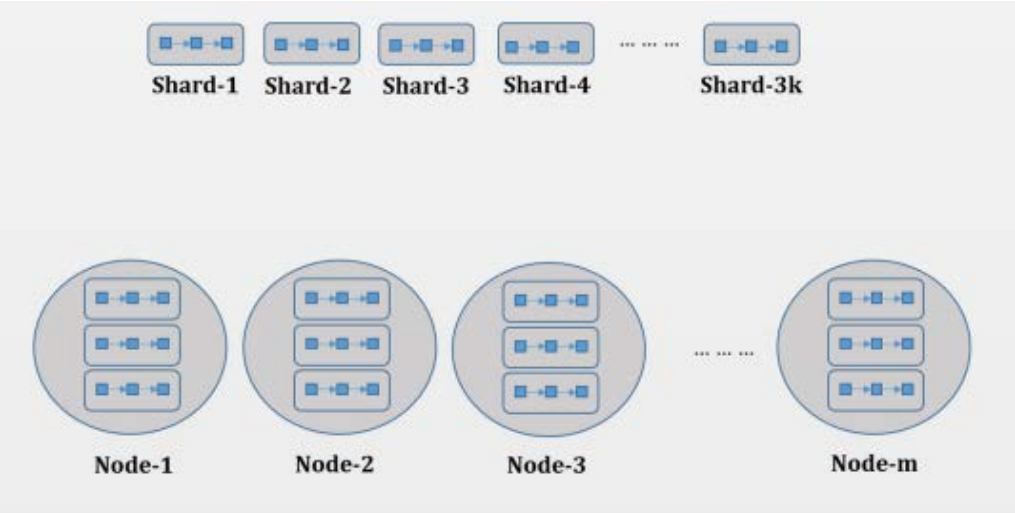

Figure 5.4: After distributing shards to nodes with overlap

for p = 3. As we place n ∗ k shards at m nodes with each node containing p shards, we can say that the following relation holds:

$$n * k = m * p \quad (5.1)$$

Collect or estimate the value of m and take input the possible values of n, p. Then, we calculate k and create the shards that we need to distribute. We keep distributing these shards as long we have them available. For the last node, we just assign all the remaining shards there.

6. RESULT AND DISCUSSION

In this section, analyse the result of STCChain with overlapping data distribution. STCChain decentralizes decryption key and uses the SSS algorithm to split key into n shares (t is the threshold). Assuming that DU has collected at least t correct key fragments, the decryption key can be reconstructed. The DU passes the identity verification and holds at least t correct key fragments, the key can be successfully reconstructed, and the data can be successfully decrypted; if the DU passes the identity verification but the number of correct key fragments held is less than t, the key cannot be reconstructed, and the decryption fails. If the DU fails to pass the identity verification, the key fragment cannot be obtained, and even if she or he obtains the encrypted data, the decryption will fail due to lack of a decryption key. The overlapping data distribution method used to reduce the storage requirement of BC. Mathematical analysis shows that, this method can not only reduce the storage requirement greatly but also ensure fault tolerance based on the selected parameters.

This project's worked in python platform .BC is produced and that is used to store data. The key processed by SSS algorithm. Overlapping data distribution method is also used to reduce the storage requirement.

Python: Python is a high-level, interpreted, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. Python is dynamically-typed and garbage-collected. Here we are using the python 3.864 version.

Blockchain: First import hash library .That is sha256 algorithm is used to produce the hash values. Preparing blocks, each blocks contains data, previous hash value and it's own hash value. Generate the genesis block, which is the first block that doesn't contain any information. The output of the blockchain in python program is shown in figure 6.1.The genesis block does not contain any data. The second block ,third block and fourth block contains the transaction details clearly.

```
Data 1: Genesis Block - 0
Hash 1: 39331a6a2ealcf31a5014b2a7c9e8dfad82df0b0666e81ce04cf8173cc5aed3e

Data 2: George sends 3.1 GC to Joe - Joe sends 2.5 GC to Adam - 39331a6a2ealcf31
a5014b2a7c9e8dfad82df0b0666e81ce04cf8173cc5aed3e
Hash 2: 98cf363aecb33989aea0425a3c1287268bd86f63851bc08c0734a31db08506d5

Data 3: Adam sends 1.2 GC to Bob - Bob sends 0.5 GC to Charlie - 98cf363aecb3398
9aea0425a3c1287268bd86f63851bc08c0734a31db08506d5
Hash 3: 6f1cfcc3082488b97db8fdf8ed33f9ac7519be3e285a37a6fcc2f1904f373589

Data 4: Charlie sends 0.2 GC to David - David sends 0.1 GC to Eric - 6f1cfcc3082
488b97db8fdf8ed33f9ac7519be3e285a37a6fcc2f1904f373589
Hash 4: 869df2f03c9860767d35b30a46233fbeea89a3000ae5019d1491e3829d1ab929
```

Figure 6.1: Blockchain output

SSS: Shamir's secret sharing algorithm is implemented in the python.In sharing scheme setting the values of threshold value (t) is 3,and the total number of shares(n) is 5. Also setting the secret. Phase I: Generation of shares by using polynomial. Phase II: Secret Reconstruction, Picking t shares randomly for reconstruction. Combining shares: when the minimum number of shares can produce the secret. Finally reconstruct secret. The output of SSS algorithm in python program is shown in below.

```
Original Secret: 1234
Shares: (92567, 765348724127404), (35907, 115162257883024), (60230, 324021921312
844), (30147, 81178621552144), (7028, 4412133204166)
Combining shares: (35907, 115162257883024), (7028, 4412133204166), (92567, 76534
8724127404)
Reconstructed secret: 1234
>>>
```

Figure 6.2: SSS output

Figure 6.2: SSS output From figure 6.2 ,it clearly shows that the original secret is 1234.There five shares are produced. The minimum number of shares is 3,which is needed to the reconstruction of scret .Output of SSS algorithm with less than minimum number of shares(t) is given in figure 6.3.The output less than t shares cannot produce the original original secret. Here two shares are combined, which less than t.

```
Original Secret: 1234
Shares: (98077, 328263407508682), (83034, 235288414439038), (96153, 315510531176
434), (23071, 18164768852062), (9919, 3357749291038)
Combining shares: (23071, 18164768852062), (98077, 328263407508682)
Reconstructed secret: -77218076419608
```

Figure 6.3: SSS output

6.1 STCCHAIN

The blockchain is added to the SSS process we get the STCChain. The blockchain is produced for the data storage. The secret is dividing into parts by using SSS algorithm. Then the parts are stored in BC. For the reconstruction of the secret minimum number of shares are used. The output of STCChain shown in figure6.4.Secret is shared and stored in blocks. The original secret is 39331,four shares are produced. The shares are stored in blocks.

```
Original Secret: 39331
Shares: (70692, 2581567205), (92972, 3395188245), (56306, 2056219257), (44242, 1
615666105)
Data 1: Genesis Block - 0
Hash 1: 39331a6a2ealcf31

Hash 1: a5014b2a7c9e8dfa

Hash 1: d82df0b0666e81ce

Hash 1: 04cf8173cc5aed3e

Data 2: (70692, 2581567205) - 39331a6a2ealcf31
Hash 2: ec06b68454131b8b

Hash 2: d55d95eadbc4950a

Hash 2: 604481bebc117308

Hash 2: b114337e7fd61329

Data 3: (70692, 2581567205) - a5014b2a7c9e8dfa
Hash 3: 5516199f4be36318

Hash 3: fad38b83a0ba08e8

Hash 3: 355ee70f253e932d

Hash 3: 73a3ddb636efbd4e
```

Figure 6.4: STCChain output (secret,shares, blocks)

Reconstruction of original secret cant possible with less than ’t’ shares. That is here t is 3 and only 2 shares are provided. So that these shares can’t produce the private key(original secret) without error. The output of recovered private key is error is given in fgure 6.5.

```
Data 6: (53989, 32866240341803) - aea120b25b601419c592c8e4bbb68f15bfd02a6354dc47
53fa4a7fb7335e42fc
Hash 6: 04439fa4d770905dbf35a04504160df818681e370f28faf25be97bf6e497a0fe

Data 7: (63633, 45656307574487) - 04439fa4d770905dbf35a04504160df818681e370f28fa
f25be97bf6e497a0fe
Hash 7: 9b0040e15395358b5ddcadladfbb3d38747a0f1dbb84664e8458235bdb0a64f3

Combining shares: (56938, 36554655013967), (23970, 6478943884919)
Reconstructed public key: -15388163484751
recoverd private_key -15388163488353
```

Figure 6.5: STCChain output(less than t shares)

Reconstruction is also correct with 't' shares. That is here t is 3.So that three shares can produce the private key(original secret) without error. The output of recovered private key is given in fgure 6.6.

```
Data 3: (58239, 3059437392299) - 61af3e4bef6c9c9751e02bdf13e43a32395b4ec5692ddcd
22100a52e78f86383
Hash 3: 58a87bfad59b66d4af498f6c389744f9b33216a96b5ac979163eb75fbea387ab

Data 4: (44066, 1751553207413) - 58a87bfad59b66d4af498f6c389744f9b33216a96b5ac97
9163eb75fbea387ab
Hash 4: 7558bdb715cf999fa9aaada68bec34ab426c77bb6d2e34964947cf2f282de01c

Data 5: (61641, 3427304606363) - 7558bdb715cf999fa9aaada68bec34ab426c77bb6d2e349
64947cf2f282de01c
Hash 5: 39521369334f09f62ee08afb348eae0615da161b4de51f22dbd0949a12b76c6e

Data 6: (32527, 954349567851) - 39521369334f09f62ee08afb348eae0615da161b4de51f22
dbd0949a12b76c6e
Hash 6: 2b9c99342aeda49b3716bae676a1881656ab2efcba8402afe2da920f36651253

Combining shares: (44066, 1751553207413), (32527, 954349567851), (40007, 1443740
087051)
Reconstructed public key: 36749
recoverd private_key 39331
```

Figure 6.6: STCChain output (with t shares)

6.2 STCChain with overlapping data distribution

STCChain with overlapping data distribution provides the high security and protect the secret. The STCChain is added with overlapping data distribution ,reduce the storage requirement in the blockchain.The main problem in a BC is the storage requirement.BC stores whole list of transactions, so the storage space increases.In blockchain nodes are the connected computers ,during transaction loss is minimum .That is high storage requirement is needed.In this method divide the whole blockchain into some shards where each shard can contain one or more blocks from the blockchain. Here we consider the division into parts.Taking the multiple copies then, we distribute the shards across the nodes in the blockchain network.So that reduce the storage This approach not only improves the storage requirement but also ensures the integrity of the data in blockchain in case of node

failures.Dividing the blockchain into parts is no easy job. We need to make sure that we keep one block in one shard only. However, one shard can contain many blocks .

The figure 6.7 is output of STCChain with overlapping data distribution.Original secret is 39331.Four shares are produced.Multiple copies are produced for the fault tolerance. First, gather information about the number of nodes or make estimation if needed.

```
Original Secret: 39331
Shares: (70692, 2581567205), (92972, 3395188245), (56306, 2056219257), (44242, 1
615666105)
Data 1: Genesis Block - 0
Hash 1: 39331a6a2ealcf31

Hash 1: a5014b2a7c9e8dfa

Hash 1: d82df0b0666e81ce

Hash 1: 04cf8173cc5aed3e

Data 2: (70692, 2581567205) - 39331a6a2ealcf31
Hash 2: ec06b68454131b8b

Hash 2: d55d95eadbc4950a

Hash 2: 604481bebc117308

Hash 2: b114337e7fd61329

Data 3: (70692, 2581567205) - a5014b2a7c9e8dfa
Hash 3: 5516199f4be36318

Hash 3: fad38b83a0ba08e8

Hash 3: 355ee70f253e932d

Hash 3: 73a3ddb636efbd4e
```

Figure 6.7: STCChain with overlapping data distribution(1)

multiple copies of the blockchain to ensure fault tolerance. Then, put a certain number of shards at each node in the network and thereby reduce the amount of storage. The figure 6.8 is the output of the program. The overlapping method is added in the blockchain python program. Same main program is used. As i mentioned earlier reconstruction is correct with ’t’ shares. That is here t is 2.So that two shares can produce the private key(original secret) without error. Also reconstruction of original secret can’t possible with less than ’t’ shares. That is here t is 2 and only lesser shares are provided. So that these shares can’t produce the private key(original secret) without error. The data on the blockchain is always increasing and so is the number of blocks. We use a dynamic method for adding these new blocks to the already existing shards which will be easier to explain with a real example. Let us consider that we are allowing 10 blocks per shard. We already have 80 blocks in the system and thus 8 shards in total. Now, when a new block arrives, we need a new shard. So, we will create a new shard and mark this shard as the shard in use. Initially, this shard will have 9 empty slots which will be filled up by later blocks. Once a new shard is created, it is given to a certain

```
Data 16: (44242, 1615666105) - 0b8d5a8b40c148e0
Hash 16: e8f2df7590947ca8

Hash 16: a18d8053917bba2f

Hash 16: 9281a727e3f2214d

Hash 16: 1961c4f4aa379c7a

Data 17: (44242, 1615666105) - 36cee0e90fdad91e
Hash 17: 6575622bb867bb42

Hash 17: d4b45b8695b50bc5

Hash 17: edbec342c6248f1c

Hash 17: 442459dae4c49746

Combining shares: (56306, 2056219257), (92972, 3395188245)
Reconstructed public key: 36749
recoverd private_key 39331
```

Figure 6.8: STCChain with overlapping data distribution(2)

number of nodes in the system. To determine which nodes will get the new shard, we use simple round-robin method. Our method was easier to use in private blockchains because of the prior knowledge on the number of nodes in the system. However, in case of public blockchains, we need to make an assumption on the number of nodes. In our opinion, there is not exact value of number of shards, shards per node, and copies of each shard that will produce the best results for all systems.

CONCLUSION

The blockchain technique which is a distributed solution of the trust problem without any third party, is a promising substitution. Moreover, blockchain is a shared database, and the data or information stored in it are characterized as ”unforgettable”, ”remain trace”, ”traceable”, ”open and transparent” and ”collectively maintained”, so it can transform industries by enabling anonymous and trustful transactions in a decentralized and trustless environment. We propose a threshold encryption protection scheme for data protection based on blockchain: STCChain solves the security and privacy issues caused when the key used for data encryption and decryption is directly stored and managed by users or third-party organizations. The experimental results show that STCChain with overlapping data distribution can not only effectively prevent attackers from stealing data illegally but also protect the privacy of keys. Overlapping data distribution method reduce the storage requirement in blockchain.

FUTURE WORKS

•Implementation of the proposed scheme in a real-world IIoT setting

•Implementing distributed data storing mechanisms in private blockchain framework

REFERENCES

[1] A. Bahga,V. K. Madisetti, and Diane Donovan , “Blockchain platform for industrial internet of things,” Journal of Software Engineering and Applications, vol. 9, no. 10, pp. 533–546, 2016

[2] L. Tan, K. Yu, C. Yang, and A. K. Bashir, “A blockchain-based Shamir’s threshold cryptography for data protection in industrial internet of things of smart city”,in Proc. 1st ACM MobiCom Workshop on Artificial Intelligence and Blockchain Technologies for Smart Cities with 6G.

[3] M. Ma, D. He, N. Kumar, K.K. R. Choo,and J. Chen, “Certificateless searchable public-key encryption scheme for industrial internet of thingss,”IEEE Transactions on Industrial Informatics, vol. 14, no. 2, pp. 759–767, feb 2018

[4] Y. Lu, J. Li, and Y. Zhang,,M. Arifuzzaman,Z. Wen, D. Zhang, andT. Sato, “Privacy-Preserving and Pairing-Free Multirecipient Certificateless Encryption With Keyword Search for Cloud-Assisted IIoT,” IEEE Internet of Things Journal, vol. 7, no. 4, pp. 2553–2562, Apr. 2020.

[5] M. Dai, S. Zhang, and H. Wang, ”A low storage room requirement framework for distributed ledger in blockchain”,IEEE access 2018.

[6] A. Dorri, S. S. Kanhere, and R. Jurdak and Diane Donovan “MOF-BC: a memory optimized and flexible blockchain for large scale networks,” Future Generation Computer Systems, vol. 92, pp. 357–373, 2019

[7] R. K. Raman , “Distributed storage meets secret sharing on the blockchain,” in Proceedings of Information Theory and Applications Workshop (ITA), San Diego, CA, USA, February 2018 30

[8] D. Mechkarosk, H.L.Wu, andL.S. Chen ,”Light Repository Blockchain System with Multisecret Sharing for Industrial Big Data”, Hindawi Security and Communication Networks Volume 2019,

[9] https://www.customlogocases.com/blog/b2b-blockchain/

[10] A. Shamir,“How to share a secret” Communications of the ACM.vol. 22, no. 11, pp. 612-613, 1979.

[11] Popovska-Mitrovikj and Daniela Mechkarosk, ”Algorithm for Reducing Storage in Blockchain based on Secret Sharing.” 2020 International Conference on Electrical, Communication, and Computer Engineering (ICECCE).IEEE, 2020.

[12] Yu K, Tan L, Yang C, Choo KK, Bashir AK, Rodrigues JJ, and,Sato T, “ A blockchain-based shamir’s threshold cryptography scheme for data protection in industrial internet of things settings”. IEEE Internet of Things Journal. 2021.

[13] http://www.springer.com/series/7410

[14] http://www.recentscientific.com/techniques-reducing-storage-blockchain

[15] https://www.statista.com/statistics/647523/worldwide-bitcoin-blockchain-size/

[16] R. K. Raman and L. R. Varshney, “Dynamic distributed storage for blockchains,” in Proc. emph IEEE Int. Symp. Inf. Theory (ISIT), Jun. 2018,

[17] Y. Kim, R. K. Raman, L. R. Varshney, andN. R. Shanbhag, . “Efficient local secret sharing for distributed blockchain systems,” ” IEEE Communications Letters

[18] M. Crosby, Nachiappan, P, Pattanayak, V. Kalyanaraman ”Blockchain technology: Beyond bitcoin”, Applied Innovation Review, Berkeley, June 2016.

[19] Muhsina K and Padmam Gopinath Kaimal , ”Techniques for reducing storage in blockchain” International Journal of Recent Scientific Research Vol. 13, Issue, 06, June, 202.