Industrial Engineering Journal
ISSN: 0970-2555
Volume : 49, Issue 8, July-December: 2020

BLOCKCHAIN-ENABLED SECURE DATA SHARING WITH GRANULAR ACCESS CONTROL FOR PERSONAL INFORMATION

Dr. BTR Naresh Reddy [1] , B. Bhagyalaxmi [2] , A. Mallesh [3]

[1] Professor, Department of Computer Applications, Aurora's PG College (MBA), Uppal, Hyderabad
Email: reddynareshbtr@gmail.com

[2] Assistant Professor, Department of Computer Applications, Aurora's PG College (MBA), Uppal, Hyderabad
Email: laxmibbaghya@gmail.com

[3] Assistant Professor, Department of Computer Applications, Aurora's PG College (MBA), Uppal, Hyderabad
Email: mallesh67@gmail.com

ABSTRACT

Open sharing and privacy protection are the cornerstones of data governance in the AI-driven era. As part of the current data-sharing solutions, users upload their data to the cloud server for distribution and storage. These solutions are essential for a common data-sharing management platform. However, as soon as a user uploads material to the server, they forfeit all ownership of it, making security and privacy extremely important. In the age of artificial intelligence, open sharing and privacy protection form the basis of data governance. As a prerequisite to the current data-sharing systems, users upload their data to the cloud server for distribution and storage. However, users lose all control over their data as soon as they upload it to the server, which raises severe concerns about security and privacy. In this user-centric strategy, the data owner maximizes the decentralization of the system by encrypting the sharing data and storing it on IPFS. The address and decryption key of the shared data will be encrypted using CP-ABE in accordance with the relevant access policy. Blockchain is used by the data owner to broadcast his data-related information and provide keys to data users. Only data users whose qualities meet the requirements of the access policy are authorized to download and decrypt the data. A specific data user can be revoked at the attribute level by the BSSPD without affecting other users, and the data owner has fine-grained access control over his data.

Keywords: Blockchain, Secure Data, Sharing, Granular Access Control, Personal Information

1 INTRODUCTION

The distributed public ledger system known as blockchain functions on a peer-to-peer network and stands out for being untrusted and decentralized. It is becoming more and more common in a range of applications and industries. A blockchain is a distributed network of peer nodes that records transactions in an immutable log. These nodes maintain a copy of the ledger by applying transactions that have been confirmed by a consensus process and organized into blocks with a hash connecting each block to the one before it. A blockchain's typical structure is shown in Figure 1. Figure 1: A blockchain's basic architecture The central nervous system of a blockchain network is a distributed ledger, which records every network transaction. Cryptographic techniques are also used in a blockchain to guarantee append-only data, meaning that once a transaction is added to the ledger, it cannot be removed. This immutability feature makes it easy to confirm that data hasn't been changed since the event. The Bitcoin currency is the first and most well-known use of the blockchain, but Ethereum adopted a different strategy by incorporating many of the essential features of Bitcoin along with smart contracts to create a platform for distributed applications. Ethereum and Bitcoin are examples of public permissionless blockchain technology. To put it simply, they are open networks that everyone can use to communicate anonymously. A permissionless blockchain allows almost anyone to participate, and everyone is anonymous. In order to mitigate the lack of trust, permissionless blockchains frequently employ transaction fees or a native cryptocurrency that is "mined" to balance the prohibitive costs of participating in a byzantine fault-tolerant consensus mechanism based on "proof of work".

2. RELEATED WORK

[1] **D. D. Detwiler** This study conducts a “One nations move to increase food safety with blockchain” Although the spinach on the grocery store shelf is a vibrant green and appears delectable, how can you be sure it is safe to consume? What if your retailer could verify every stop that spinach took on the way to the store, as well as where it was cultivated, handled, kept, and inspected, with absolute certainty? Blockchain, a shared, distributed ledger technology, gives your retailer access to this data. By directly integrating growers, processors, distributors, suppliers, retailers, and regulators with a common, immutable view of their transaction history, blockchain-based solutions have the potential to change the food business.

[2] **Boneh, D., Franklin, M. Identity based encryption from the Weil pairing. In: Kilian,** We suggest an identity-based encryption system that is fully operational (IBE). Under the random oracle model, the system has chosen ciphertext security while assuming a special case of the computational Diffie- Hellman problem. The foundation of our system is a bilinear map between groups. One such map is the Weil pairing on elliptic curves. We provide clear definitions for safe identity-based encryption techniques and list many uses for these systems.

[3] **Boneh, D., Boyen, X. Efficient** “elective-ID secure identity based en- cryption without random oracles“ We provide an identity-based encryption method that is 100 percent secure and whose security proof does not rely on the random oracle heuristic. The decisional bilinear Diffie-Hellman assumption underlies security. The security reduction from the underlying complexity assumption resulted in a significant penalty factor for prior designs of this sort. The current system's security reduction is polynomial across all parameters.

[4] **Boneh, D., Boyen, X. Secur “**identity based encryption without random oracles**”** Using no random oracles, we introduce the first effective Identity-Based Encryption (IBE) technique. We initially discuss our IBE construction before reducing the decisional Bilinear Diffie-Hellman (BDH) problem to describe the security of our system. Additionally, we demonstrate that a new signature scheme that is secure without random oracles under the computational Diffie-Hellman assumption can be created using our techniques.

[5] **R. S. Wahby, I. Tzialla, A. Shelat, J. Thaler and M. Walfish “**Cluster computing in

zero knowledge, EUROCRYPT”

For scalability and economic considerations, large computations that may be conducted in distributed parallel are frequently carried out on computer clusters. Similar calculations are employed in a variety of applications, including, but not limited to, statistical machine translation, webgraph mining, and machine learning. But frequently, just the output of the computation can be published because the input data is secret. In these circumstances, zero-knowledge proofs would enable the verification of the output's validity without disclosing (extra) information about the input. We examine theoretical and applied elements of zero-knowledge proofs for cluster computations in this paper. We create zero-knowledge proof systems and assess them for:

(i) A proof confirms the accuracy of a cluster computation, and
(ii) creating the proof is a cluster computation in and of itself, with a structure and complexity comparable to the original one.

We specifically concentrate on MapReduce, a beautiful and well-liked cluster computing method. A monolithic NP statement that accounts for all mappers, all reducers, and shuffling can theoretically demonstrate the soundness of a MapReduce computation using previous zero-knowledge proof techniques. Yet, it is unclear how to produce the evidence for such monolithic claims using distributed systems' parallel execution. The correctness of a cluster computation is attested to by a proof, and the cluster computation used to generate the proof shares the same complexity and structural characteristics as the original cluster computation. In particular, we concentrate on MapReduce, a sophisticated and well-liked cluster computing method. A monolithic NP statement that reasons about all mappers, all reducers, and shuffling is theoretically able to demonstrate the correctness of a MapReduce computation. This is possible since previous zero-knowledge proof systems can do this. Although a distributed system can execute these statements in parallel, it is unclear how to produce the proof for such monolithic claims.

3 IMPLEMENTATION STUDY

Block chain generator: After generating the block chain, which creates 10 block chain users and 10 block chain private keys, we connect to the Solidarity network to store the data as blocks using the master key.

User login: First, the user must register. Then, after registering, the user can upload a message and a document, which will be stored in the block chain. The user can then view the information that has been shared with them, but only authorised users will be able to see the data; everyone else will not be able to.

4 PROPOSED WORK

ABE is regarded as the most suitable solution to address issues with data security and privacy protection in a distributed setting. In order to accomplish fine-grained access control over data on the blockchain, researchers have lately adopted ABE. A decentralised access control system was presented by Jemel and Serhrouchni [26]. For the first time, researchers executed a CP-ABE algorithm on blockchain nodes to validate the legality of user access rights. SetPolicy and GetAccess are the two sorts of transactions that the scheme envisions. But, it does not make use of Smart Contracts, and it is clear that the plan cannot fulfil more

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 49, Issue 8, July-December: 2020

demanding needs. Based on ABE and blockchain, Sun et al. developed a paradigm of safe storage and efficient sharing for electronic medical data [27] that offers superior access management. Medical information about patients is encrypted using ABE and stored on IPFS by doctors. It does not, however, make use of smart contracts. It cannot perform more complex business operations; it can only broadcast some ABE parameters that are recorded in transactions. Users share secret keys in a sharing method Wang et al. suggested [28]. It understands that the owner of the data has a precise access control on that data. The recovery of ciphertext keywords is also made possible via the Ethereum Smart Contract. Nevertheless, many off-chain communications are necessary between users, and more critically, the permit revocation is not implemented. To record and retain medical data, Pournaghi et al. suggested MedSBA, a safe and effective sharing system based on blockchain and ABE [29]. By broadcasting a new strategy to cover the previous transaction, it implements the update and revocation of permissions, however users who do not want their rights revoked will be forced to update their keys.
ADVANTAGES OF THE PROPOSED SYSTEM:
1. HIGH ACCURACY
2. STRONG EFFICIENCY

5 METHODOLOGIES AND ALOGRITHAM

Elliptic Curve Cryptography (ECC) is a contemporary family of public-key cryptosystems that is based on the algebraic structures of elliptic curves over finite fields and on the challenge of the Elliptic Curve Discrete Logarithm Problem (ECDLP).ECC implements encryption, signatures, and key exchange, which are the three main asymmetric cryptosystem features.
Since ECC utilises fewer keys and signatures than RSA for the same level of security and offers very quick key generation, quick key agreement, and quick signatures, it is seen as the logical modern replacement for the RSA cryptosystem.
ECC Keys:
The ECC uses integer private keys that fall inside the field size range of the curve, which is typically 256 bits. The following is an example of a 256-bit ECC private key in hexadecimal format: 0x51897b64e85c3f714bba707e867914295a1377a7463a9dae8ea6a8b914246319.
ECC cryptography is incredibly quick because the key generation is as easy as securely producing a random integer within a given range. An ECC private key that falls inside the range is valid.
The public keys in the ECC are what are known as EC points, which are pairs of x, y-coordinates. EC points can be compressed to just one coordinate plus one bit because of their unique characteristics (odd or even). The resulting 257-bit integer is the compressed public key, which corresponds to a 256-bit ECC private key.
0x02f54ba86dc1ccb5bed0224d23f01ed87e4a443c47fc690d7797a13d41d2340e1a
is an example of an ECC public key (equivalent to the above private key, encoded in the Ethereum format, as hex with prefix 02 or 03). The public key in this format requires 33 bytes (66 hex digits), which can be compressed to exactly 257 bits.

Curves and key length

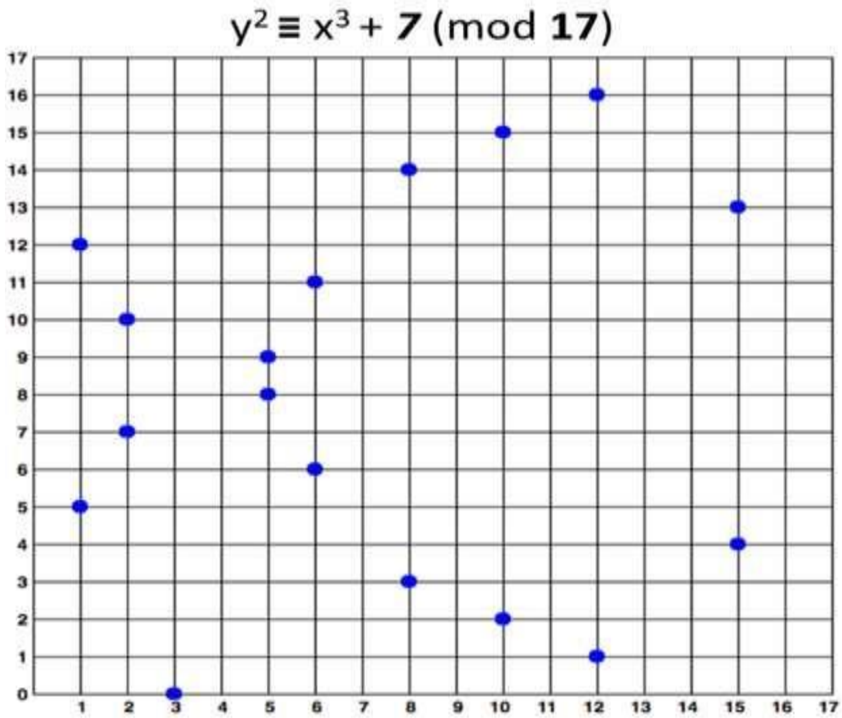
Industrial Engineering Journal
ISSN: 0970-2555
Volume : 49, Issue 8, July-December: 2020

Several underlying elliptic curves can be used with ECC cryptographic techniques. Various curves offer various levels of security (cryptographic strength), performance (speed), and key length, as well as perhaps involving various methods. In addition to having a name (named curves, for example secp256k1 or Curve25519), a field size (which defines the key length, for example 256 bits), security strength (typically the field size / 2 or less), performance (operations/sec) and many other parameters, ECC curves are widely used in cryptographic libraries and security standards. The length of ECC keys is closely related to the underlying curve. The default key length for the ECC private keys in the majority of programmes (including OpenSSL, OpenSSH, and Bitcoin) is 256 bits, but many different ECC key sizes are feasible depending on the curve: 192-bit (curve secp192r1), 233-bit (curve sect233k1), 224-bit (curve secp224k1), 256-bit (curves secp256k1 and Curve25519), 283-bit (curve sect283k1), 384-bit (curves p384 and secp384r1), 409-bit (curve sect409r1), 414-bit (curve Curve41417), 448-bit (curve Curve448-Goldilocks), 511-bit (curve M-511), 521-bit (curve P-521), 571-bit (curve sect571k1) and many others. The elliptic curve cryptography (ECC) employs elliptic curves over either a field of infinite size, Fp (where p is prime and p > 3), or F2m (where p = 2 m_). This indicates that the curve's points can only have integer coordinates within the field, which is a square matrix of size p x p. Every algebraic operation performed on the field (such as point addition and multiplication) yields a new point. The modular form of the elliptic curve equation over the finite field Fp is as follows:

- y2 ≡ x3 + _**a**_x + ***b*** (mod ***p***)

Thus, the secp256k1 "Bitcoin curve" looks like this:

- y2 ≡ x3 + ***7*** (mod ***p***)



The blue dots in the accompanying figure make up the elliptic curve over a finite field called y2 x3 + 7 (mod 17). In other words, the "elliptic curves" employed in cryptography are actually "sets of points in square matrices" rather than traditional "curves."

The curve seen above is "educational." It offers relatively short key lengths (4-5 bits). Developers generally employ curves with 256 bits or greater in the real world.

Calculations for Elliptic Curves over Finite Fields

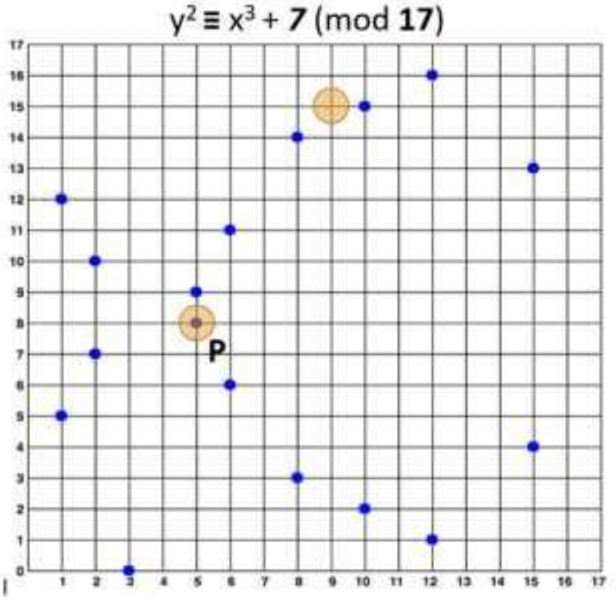
Calculating whether a given point over a finite field conforms to a given elliptic curve is quite simple. As an illustration, a point x, y is only a part of the curve y2 x3 + 7 (mod 17) if and only if:

- x3 + **7** - y2 ≡ 0 (mod **17**)

Because (5**3 + 7 - 8**2)% 17 == 0, the point P 5, 8 is on the curve. The curve does not include the point (9,15),

because (9**3 + 7 - 15**2)% 17!= 0. The computations used here are done in Python. The points 5, 8, and 9 in

relation to the elliptic curve given above are shown below:



ECC Point Multiplication by Integer

An additional point can be obtained by adding two elliptic curve (EC) points. The addition of EC points is

the name of this process. G + G = 2 * G is the consequence of adding a point G to itself. The next time we add

G to the outcome, we will get 3 * G, and so on. This is the definition of EC point multiplication.

The outcome of multiplying an integer k by a point G on an elliptic curve over a finite field (EC point) is

another EC point P on the same curve, and this operation is quick:

- **P** = **k** * **G**

For the sake of simplicity, we will omit the formulas and transformations used in the aforementioned

process. Knowing that multiplication an EC point by an integer results in another EC point on the same

curve and that this operation is quick is crucial. An exclusive EC point known as "infinity" is produced

when multiplying an EC point by 0.

The Wikipedia article on EC point multiplication is open to everyone.For instance, multiply EC Point by an

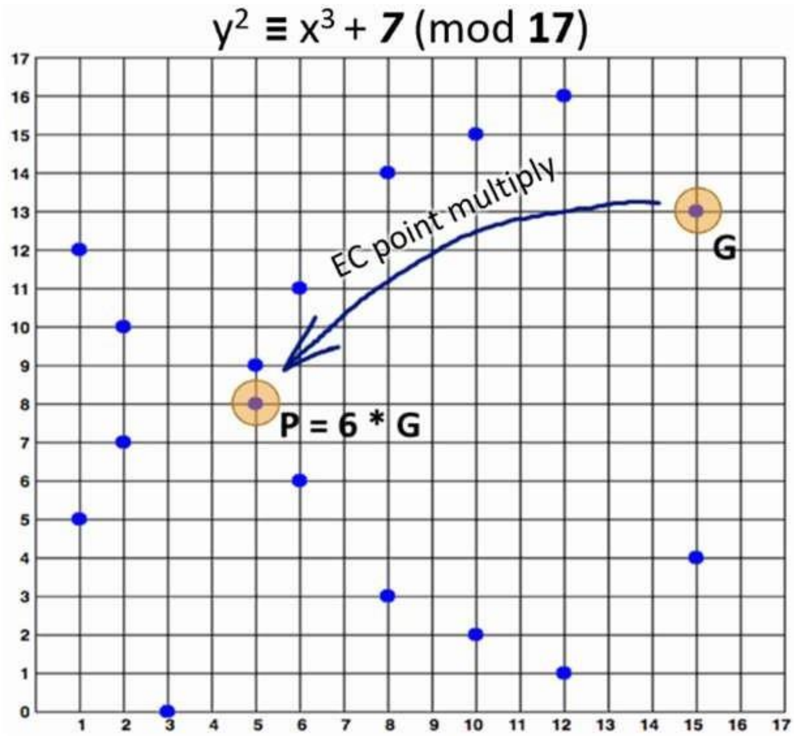
integer.The EC multiplication formulas vary depending on how the curve is

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 49, Issue 8, July-December: 2020

represented. We will use an
elliptic curve in the standard Weierstrass form for this example.
As an illustration, let's multiply the EC point G = 15, 13 by k = 6 on the elliptic curve over the finite field y2
x3 + 7 (mod 17). We will receive an EC point P = 5, 8 as follows:

- **P** = **k** * **G** = **6** * {**15**, **13**} = {**5**, **8**}

This illustration of EC point multiplication is shown below:



Industrial Engineering Journal
ISSN: 0970-2555
Volume : 49, Issue 8, July-December: 2020

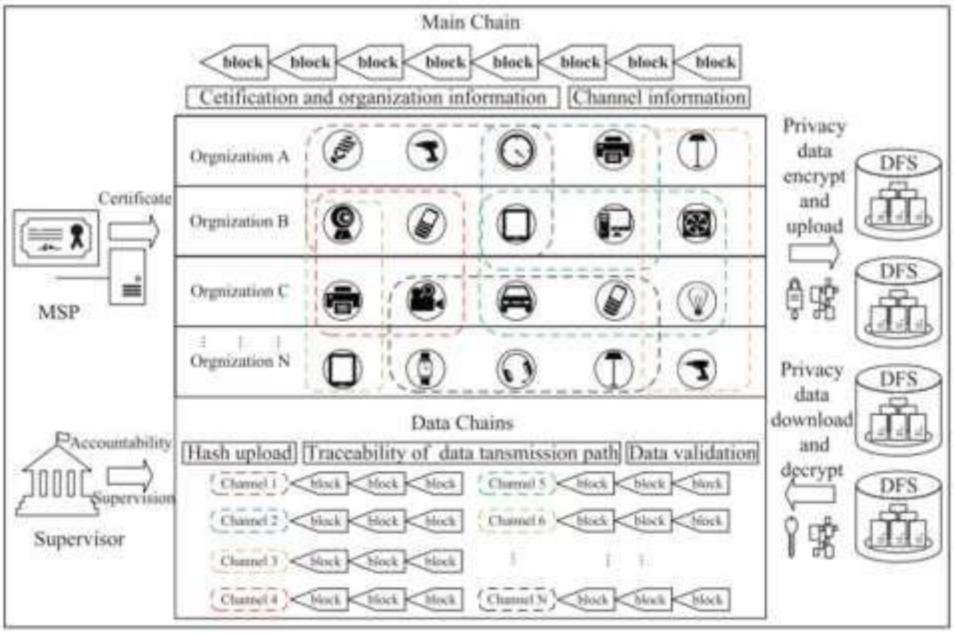


Fig. 1: propose Architecture

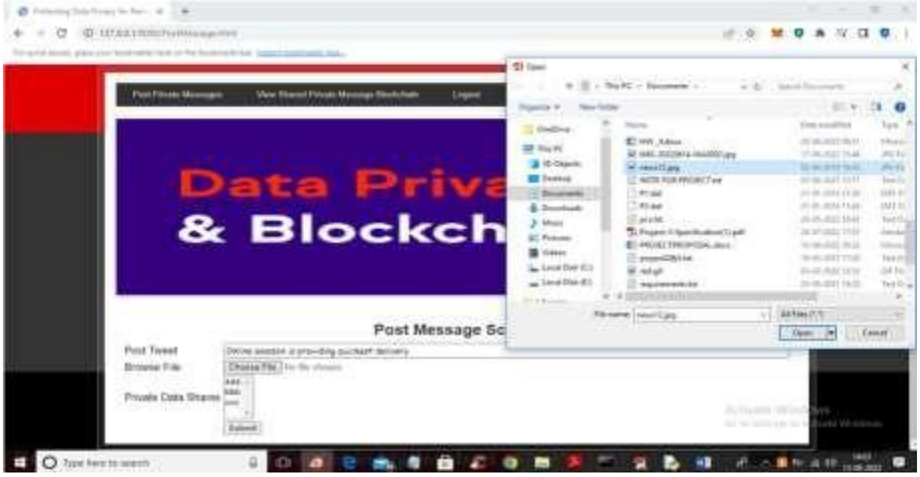
6 RESULTS AND DISCUSSION

To create an account, click the link that says "New User Registration Here" in the screen above.

User enters sign-up information on the screen above, clicks submit, and the following output appears.

After completing the above screen's user signup and saving their information to Blockchain, click the "User Login" link to access the following screen.

The user is logging in on the screen above, and then they see the screen below.



The user types a message, uploads an image, and then chooses which users to share it with. You can choose several users by holding down the CTRL key, as shown in the screen below.

In the screenshot above, user John is publishing a post and subsequently sharing it with users "aaa" and "bbb," leaving user "ccc" unable to see the post. After pressing the "Submit" button to preserve the post in Blockchain, the user receives the below result.

7. CONCLUSION AND FUTURE WORK

We have proposed an improved delicately scheme on top of non-transactional conditions in permission blockchain to further illustrate the privacy. Our method may hide the information by transforming the plaintext into the cipher text without the usage of cutting-edge technology like ring signature, homomorphism encryption, or zero-knowledge proofs. Our method is practical, effective, and beneficial for applications. It also removes the difficult certificate issuance and management included in the traditional PKI system and provides a high level of security that can prevent passive and disguised attacks. In many applications for non-transactional situations, this technology provides a novel way to preserve crucial transaction confidentiality.

REFRENCES

[1] The Linux Foundation Helps Hyperledger Build the Most Vibrant Open Source

Ecosystem for Blockchain. http://www.linuxfoundation.org/.
[2] S. Omohundro. Cryptocurrencies, smart contracts, and artificial intelli- gence. AI Matters, 1(2):19C21, Dec. 2014
[3] D. D. Detwiler. One nations move to increase food safety with blockchain. https://www.ibm.com/blogs/blockchain/2018/02/one-nations- move-to-increase-food-safety-with-blockchain/,2018. [Online; accessed 1-May-2018].
[4] Shamir, A. Identity-based cryptosystems and signature schemes. In: Blakely, G.R., Chaum, D. (eds.) CRYPTO 1984. LNCS, vol. 196, pp. 47C53. Springer, Heidelberg (1985)
[5] Boneh, D., Franklin, M. Identity-based encryption from the Weil pairing. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 213C229. Springer, Berlin, Ger- many (2001)
[6] Boneh, D., Boyen, X. Efficient selective-ID secure identity based en- cryption without random oracles. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 223C238. Springer, Berlin, Germany (2004)
[7] Boneh, D., Boyen, X. Secure identity based encryption without random oracles. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, Springer, Berlin, Germany (2004).
[8] Gentry, C. Practical identity-based encryption without random oracles. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 445C464. Springer, Berlin, Germany (2006).
[9] Labs, Shen NoetherMrl. Ring confidential transactions. 2016.
[10] R. S. Wahby, I. Tzialla, A. Shelat, J. Thaler and M. Walfish. Doubly- Efficient zkSNARKs Without Trusted Setup. 2018 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, 2018, pp. 926-943.
[11] B. Bnz J. Bootle D. Boneh A. Poelstra P. Wuille G. Maxwell. Bullet- proofs: Efficient range proofs for confidential transactions", IEEE S&P May 2018.
[12] A. Chiesa E. Tromer M. Virza. Cluster computing in zero knowledge, EUROCRYPT Apr. 2015.
[13] A. Chiesa M. A. Forbes N. Spooner. A zero knowledge sumcheck and its applications. CoRR abs1704.02086 2017.