CS6290: Reading Summary 2

LING Shengchen
Dept. of Computer Science
ID: 57730900

I. Summary of Paper [1]

A. Problem Statement

The paper mainly focuses on two series of problems.

First, new decentralized DNS and PKI can be built on blockchains, but not much production experience is available for design tradeoff guidance, in respect of security, reliability, and deployment of consensus-breaking changes.

Second, technical limitations of contemporary blockchains prevent naming systems from going further, which are 1) data in one individual block are limited [2], 2) latency of transaction confirmation results in low writing and updating rate [3], 3) node bandwidths limit the operations of naming systems [4], and 4) new nodes need to take a long time to audit the whole ledger independently.

B. Problem Significance

Blockchains and their respective P2P networks has been developing rapidly in the past few years. Decentralized applications and services based on blockchains can minimize the trust that users have to put in one single party, e.g., a certificate authority (CA), and many non-financial applications are eager to a naming system that binds human-readable names with corresponding values.

C. State of the Art

The paper uses "naming system" to denote that 1) names are human-readable, 2) name-value pairs can be owned by cryptographic key pairs, and 3) no central trusted party. Satisfying these three properties simultaneously was considered impossible [5], but wild discussions have been conducted.

Namecoin did so based on blockchain in PGP-like format [6], however, some problems are proposed in this paper, see section D. UIA [7] chooses only to provide locally unique names, while InCommon [8] and OpenID [9] set a federation to manage the bindings as authentication systems.

Some other blockchains like Ethereum [10] and Bitshares [11] also support naming systems, as well as sidechains [12], but they are not as secure as Bitcoin. Non-blockchain based systems like Keybase [13] and CONIKS [14] achieve similar goals, but still remain trust transfer.

D. Contributions

1) *From the perspective of production experiences.* The paper proposed a series lessons learnt based on the experiences in deploying and operating a decentralized PKI service.

First, there usually exists a tradeoff between security and new functionalities. The paper discovers a potential 51% attack security problem [15] that one mining pool owns over 51% of total computational power of Namecoin network, where the previous naming system was built. Second, the paper analyzes the networking issues and transaction pending on Namecoin network, holding a conclusion that the more actively a blockchain is maintained with more financial incentives, the more secure and reliable the blockchain will be. Third, selfish mining is not just in theory [16], but similar behaviors can be detected in production data. Fourth, consensus-breaking changes are hard not only in engineering problems, but also incentive structures. Fifth, merged mining [17][18] fails in practice as no sufficient computational cycles dedicated to supporting multiple blockchains.

Thus, the paper concludes that decentralized applications (including the PKI system) should be implemented on the largest and most actively maintained blockchain (which refers to Bitcoin at present).

2) *From the perspective of novelty of technical solution.* To solve the challenges mentioned above, the paper proposes Blockstack on top of blockchains.

The Blockstack structure logically contains two planes and four layers, which are control plane (consists of blockchain layer and virtualchain layer) and data plane (consists of routing layer and storage layer). The former is for human-readable names and its key pair bindings, while the latter is for data storage and availability, see Fig.1.

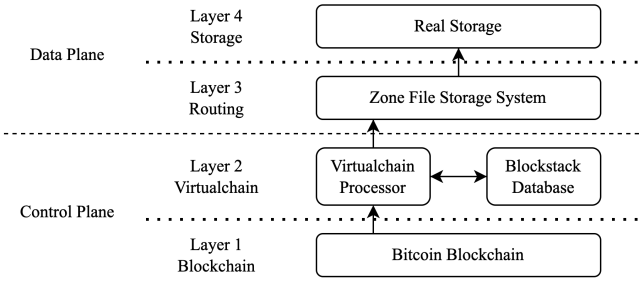

Fig. 1. Blockstack structure

The key contribution of Blockstack is the *virtualchain*, which constructs a global-state state machine after processing the retrieved information from blockchain and implements changes simply by introducing new types of state machines instead of changing the whole blockchain.

Blockstack also enables new nodes of quick start by introducing checkpoints and skip lists, aiming to reduce the data quantity that new nodes must audit. The process is called Simplified Name Verification (SNV), which is to compare and verify the derived final consensus hash of an untrusted database state with a trusted one on the same block height.

3) *From the perspective of positiveness of experimental evaluation result.* The performance of reading and writing acts at competitive rates, which is similar to direct accessing, with only a small CPU overhead. However, optimizations still remains to be implemented.

E. Remaining Questions

1) The processing rate in Blockchain layer is limited by the performance of Bitcoin, thus the experimental results can be optimized if switching to other blockchains, e.g., Ethereum.

2) The production data of 33,000 entries and 200,000 transactions seem not sufficient enough to cover all circumstances, more real data are needed.

3) The detailed pricing and exchanging mechanism is out of the scope of this paper, which can deeply affect the long-time operation of the naming system.

II. SUMMARY OF PAPER [19]

A. Problem Statement

The paper targets the problem of privacy on blockchains. Particularly, how to guarantee the privacy of transaction information despite they are stored in a public ledger.

B. Problem Significance

Cryptocurrencies including Bitcoin have achieved widespread adoption in recent years. To protect on-chain privacy, ordinary users need a system to guarantee that their privacy data on blockchains, e.g., transaction habits and account balances, are not publicly accessible by others. Hackers may analyze the on-chain data to bind addresses with identities in real life, thus anonymity is compromised.

Moreover, the value of a coin needs to be guaranteed independent of its transaction history. Theoretically, fungible tokens share roughly the same values, but some owners or suspicious transferring behaviors in history may influence token pricing.

C. State of the Art

Zerocoin [25] has addressed some of the privacy issues mentioned above and provided strong anonymous guarantees, but the transactions destinations and amounts are not under protection. Moreover, Zerocoin requires double-discrete-logarithm proofs of knowledge, which results in its low performance, and has some critical functional limitations, e.g., its coins are in fixed denominations, and cannot be directly paid in the form of "zerocoins".

Pinocchio Coin [26] addressed some of the issues of Zerocoin using *zero-knowledge Succinct Non-interactive ARguments of Knowledge* (zk-SNARK), but still only supports fixed-value coins and is more like a mix. Moreover, its complexity scales linearly in the amount of coins.

N. Bitansky et al. proposed a formal definition of zk-SNARK [20], and many zk-SNARK constructions have been proposed especially in the aspect of quadratic arithmetic programs [21][22][23][24]. Among them, B. Parno et al. [22] presented for programs without data dependencies, E. Ben-Sasson et al. [23] presented for programs with data dependencies, and E. Ben-Sasson et al. [24] presented an implementation with runtime code generation based on [22], which can reduce costs and allow universal key pairs.

D. Contributions

1) *From the perspective of novelty of the technical solution,* the paper proposes *decentralized anonymous payment schemes* (DAP schemes) to enable users to send transactions privately and directly, without revealing transaction's origin, destination, and amount. More precisely, the DAP scheme includes the following six steps.

First, it implements user anonymous with fixed coin values to hide transaction origins, like Zerocoin. Second, it compresses the coin commitments by introducing a Merkle tree, reducing the time and space complexity from linear to logarithmic. Third, it extends coins for direct anonymous transactions by modifying the derivations of coin commitments and using three pseudorandom functions $PRF_x^{addr}()$, $PRF_x^{sn}()$, $PRF_x^{pk}()$, to target transactions and derive serial numbers. Fourth, it transfers the secret values securely by modifying the key pair structure of an address, computing ciphertext C_1, and including it in the pour transaction tx_{Pour}, thus the receiver can scan in the public ledger, find it, and decrypt it, see Fig.2. Fifth, it enables redeeming coins back to the base currency (e.g., Bitcoin) by modifying the pour operation to include a public output, which contains a nonnegative v_{pub} of currency value and an arbitrary string $info$ of target address. Sixth, it prevents malleability attacks on a pour transaction tx_{Pour} by modifying the NP statement $POUR$ appending with digital signatures.

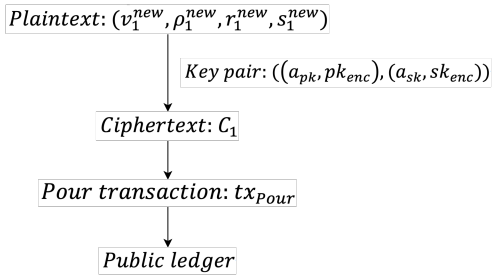

Fig. 2. Transfer secret values

2) *From the perspective of depth of theoretical analysis*, the paper proves the system's security under specific cryptographic assumptions, including ledger indistinguishability, transaction non-malleability, and balance. The paper provides formal definitions and proofs in its extended version [27].

3) *From the perspective of positiveness of experimental evaluation result*, the paper also implements Zerocash, a practical instantiation of DAP scheme, to ensure and verify that the zk-SNARK is practically efficient enough. In the experiment, zk-SNARK is mainly used to prove and verify a specific NP statement of $POUR$, by instantiating all necessary cryptographic ingredients based on SHA-256. The result shows

that a zk-SNARK prover runs in few minutes while verifier in few milliseconds.

E. Remaining Questions

1) Such privacy protection may not be accessible to regulation departments, which is more convenient to money laundry and DeFi hacking.

2) The new functionalities can be directly integrated with the base blockchain (base currency) itself, either by implementing another hard fork or by introducing new zk-Rollups and zkEVMs, e.g., zkSync we see today [28][29].

3) The main bottleneck of performance lies on the prover, where large scale of computations is made. Thus it can be optimized by split up prover's job into parallel computations [30], exploiting the power of the cloud to enable zero-knowledge proof in a more distributed way.

REFERENCES

[1] M. Ali, J. Nelson, R. Shea, and M. J. Freedman, "Blockstack: A Global Naming and Storage System Secured by Blockchains", in *Proc. of USENIX Annual Technical Conference (USENIX ATC '16)*, 2016.

[2] Satoshi Nakamoto, "Bitcoin: A peer-to-peer electronic cash system", https://bitcoin. org/bitcoin.pdf, 2008.

[3] J. Bonneau, A. Miller, J. Clark, A. Narayanan, J. A. Kroll, and E. W. Felten, "SoK: Research perspectives and chal- lenges for bitcoin and cryptocurrencies", in *IEEE Symposium on Security and Privacy*, 2015.

[4] Bitcoin transactions per blocks, https://blockchain.info/charts/n-transactions-per-block.

[5] D. Kaminsky, "Spelunking the triangle: Exploring aaron swartzs take on zookos triangle", http://dankaminsky.com/2011/01/13/spelunk-tri/, 2011.

[6] P. R. Zimmermann, "The Official PGP User's Guide", MIT Press, 1995.

[7] B. Ford, J. Strauss, C. Lesniewski-Laas, S. Rhea, F. Kaashoek, and R. Morris, "Persistent personal names for globally connected mobile devices", in *Proc. of the 7th USENIX Symposium on Operating Systems Design and Implementation (OSDI '06)*, 2006.

[8] Incommon federation, https://www.incommon.org/federation/.

[9] D. Recordon and D. Reed, "Openid 2.0: A platform for user-centric identity management", in *Proc. of the Second ACM Workshop on Digital Identity Management (DIM '06)*, 2006.

[10] A next-generation smart contract and decentralized application platform, https://github.com/ethereum/wiki/wiki/White-Paper, 2016.

[11] Bitshares namespaces, http://docs.bitshares.eu/namespaces/index.html, 2016.

[12] A. Back, M. Corallo, L. Dashjr, M. Friedenbach, G. Maxwell, A. Miller, A. Poelstra, J. Timon, and P. Wuille, "Enabling Blockchain Innovations with Pegged Sidechains", https: //blockstream.com/sidechains.pdf, 2014.

[13] Keybase, https://keybase.io.

[14] M. S. Melara, A. Blankstein, J. Bonneau, E. W. Felten, and M. J. Freedman, "CONIKS: bringing key transparency to end users", in *24th USENIX Security Symposium (USENIX Security 15)*, 2015.

[15] J. A. Kroll, I. C. Davey, and E. W. Felten, "The economics of bitcoin mining, or bitcoin in the presence of adversaries", in *The Workshop on the Economics of Information Security (WEIS)*, 2013.

[16] I. Eyal and E. G. Sirer, "Majority is not enough: Bitcoin mining is vulnerable", CoRR, abs/1311.0243, 2013.

[17] Merge mining specification. https://en.bitcoin.it/wiki/Merged_mining_specification.

[18] H. Kalodner, M. Carlsten, P. Ellenbogen, J. Bonneau, and A. Narayanan, "An empirical study of Namecoin and lessons for decentralized namespace design", in *Proc. of the 14th Workshop on the Economics of Information Security (WEIS '15)*, 2015.

[19] E. Ben-Sasson et al., "Zerocash: Decentralized Anonymous Payments from Bitcoin", *IEEE Symposium on Security and Privacy*, 2014.

[20] N. Bitansky, A. Chiesa, Y. Ishai, R. Ostrovsky, and O. Paneth, "Succinct non-interactive arguments via linear interactive proofs," in *Theory of Cryptography Conference*, 2013.

[21] R. Gennaro, C. Gentry, B. Parno, and M. Raykova, "Quadratic span programs and succinct NIZKs without PCPs," in *EUROCRYPT '13*, 2013.

[22] B. Parno, C. Gentry, J. Howell, and M. Raykova, "Pinocchio: nearly practical verifiable computation," in *IEEE Symposium on Security and Privacy*, 2013.

[23] E. Ben-Sasson, A. Chiesa, D. Genkin, E. Tromer, and M. Virza, "SNARKs for C: verifying program executions succinctly and in zero knowledge," in *CRYPTO '13*, 2013.

[24] E. Ben-Sasson, A. Chiesa, E. Tromer, and M. Virza, "Succinct non-interactive arguments for a von Neumann architecture," in *Proc. of the 23rd USENIX conference on Security Symposium*, 2014.

[25] I. Miers, C. Garman, M. Green, and A. D. Rubin, "Zerocoin: Anonymous distributed e-cash from bitcoin," in *IEEE Symposium on Security and Privacy*, 2013.

[26] G. Danezis, C. Fournet, M. Kohlweiss, and B. Parno, "Pinocchio Coin: building Zerocoin from a succinct pairing-based proof system," in *Proc. of the First ACM workshop on Language support for privacy-enhancing technologies*, 2013.

[27] E. Ben-Sasson et al., "Zerocash: Decentralized Anonymous Payments from Bitcoin (extended version)", Cryptology ePrint Archive, 2014.

[28] C. Sguanci, R. Spatafora, and A. M. Vergani, "Layer 2 Blockchain Scaling: a Survey", arXiv:2107.10881, 2021.

[29] zkSync Overview, https://docs.zksync.io/userdocs/intro/.

[30] Y. Ishai, E. Kushilevitz, R. Ostrovsky, and A. Sahai, "Zero-knowledge from secure multiparty computation", In *Proc. of ACM symposium on Theory of computing*, 2007.