# Decentralizing Privacy: Protecting Personal Data by Blockchain

**Shiv Shankar Singh**

***Abstract— the security of users are questioned when security breaches occur in data when third parties are incorporated for collecting and controlling huge amount of personal data. A decentralized network of peers are accompanied by a public ledger and it has demonstrated bitcoin in the financial space that trusted and auditable computing. This paper describes a decentralized personal data management system for ensuring users control over their data. A protocol is implemented that is capable of turning a blockchain into an automated access-control manager that is not requiring trust in a third party. There are no strict financial transactions in our system. They are used for carrying instructions like querying, storing and sharing data. Finally, some possible blockchain extensions are discussed that are able to harness them into a well-rounded solution for faithful computing problems in society.***

***Keywords— bitcoin, privacy, blockchain, personal data.***

## I. INTRODUCTION

There has been a rapid increase in the amount of data in our world. The recent study[1] reports that the past couple of years collected 20% of the world's data. After being incepted[2], around 300 petabytes of delicate data has been collected by world's largest social site Facebook. It is 100 times larger than the quantity of data collected by Library of Congress over 200 years[3]. The innovation and economic growth has resulted by the constant analysis and collection of data in this era of big data. The future trends prediction, optimization of process of corporate decision making, is done by the data collected by organizations and companies for personalizing services. Data is one of the most valuable asset of our economy[4]. The concern about user's privacy is growing with the benefits of a data-driven society. The large quantities of sensitive and private information is held by centralized organization in both private and public sectors. No individual has control over the data stored or used about them. The controversial issues related to the privacy has been reported by public media. The best known example of this issue is the large-scale scientific experiment conducted by Facebook without the knowledge of participants[5] and story about surveillance by government[6]. These privacy issues have been addresses several times, both legislatively[7] and technologically. The computations of data and answers instead to raw data[7] is presented by a recently developed framework namely, openPDS. The OAuth protocol based software for serving as centralized trusted authorities is used by leading companies for implementing proprietary authentication. The privacy concerns of a private data is a major perspective of concern for researchers and various techniques are developed for the same. The personally identifiable information is protected by data anonymization methods. K-anonymity is one of that techniques [8]. A diverse set of possible values[9] is used to protect the sensitive data by using related extensions to k-anonymity including I-diversity and the distribution of sensitive data[10] is protected by t-closeness. Differential privacy methods are included in other privacy preserving methods for perturbing data and adding "noise to computational process prior to sharing data"[11] and schemes of encryption that permit running of queries and computations over encrypted data. It is possible to run any computation on data which is encrypted by "fully homomorphic encryption" (FHE)[12]. But it is inefficient to practice it. Bitcoin was the first system of such kind which allowed the transfer of currency securely without using a centralized regulator by the use of a blockchain or publicly verifiable open ledger. The functions that need trusted auditability and computing used blockchains. What that is contributed by us:

1. The management of personal data that aims at privacy is constructed by combining blockchain and offblockchain storage.
2. A vital resource in trusted-computing is blockchains.

## II. PRIVACY PROBLEM

This paper addresses the privacy concerns faced by a user while using third party services. The main focus is mobile applications because users require to install the mobile applications for the deployment of services. A high resolution of personal data is collected by these applications without the control or knowledge of a user. Our analysis makes an assumption of honest-but-curios services (followed by protocol). Other concerns of data privacy such as sharing of patient's medical data for performing scientific research for monitoring how to use it and how to instantly opt-out from it can use the same system. The following privacy issues are protected by our system: Data Ownership. The main focus of our framework is to ensure that the users are able of owning and controlling their personal data.

**Auditability and Data Transparency:** A complete transparency is offered to the user about the kind of data collected and the way it is accessed.

**Fine-grained Access Control:** A set of permissions when signed-up is granted by users when using mobile applications. This presents a reason of concern. Opting-out from the agreement is the only way to alter the agreement

**Revised Manuscript Received on August 05, 2019.**

**Shiv Shankar Singh,** Department of Computer Science and Engineering, Sanskriti University, Uttar Pradesh,India. (E-mail: sanpubip@gmail.com)

because permissions are granted indefinitely. By using our framework, user is able to revoke earlier collected data by altering the existing set of permissions anytime. It is possible to keep same user-interface and to securely store access-control policies on a blockchain.

## III. PROPOSED SYSTEM

Beginning system's overview, the three entities comprised in this system are illustrated. The users of mobile phone who are interested in using and downloading applications; data for operational and business reasons (personalized service, targeted ads, etc); and the entities and nodes for maintaining and verifying the identity of a blockchain. The system verifies their identity by keeping the users anonymous. The designing of system is done as follows:

The two new types of transactions are accepted by blockchain:

$T_{access}$: It is used for "access control management";

$T_{data}$: It is used for "storage and retrieval of data".

A shared encryption key is used for encrypting data collected on a phone (e.g., sensor data such as location) and in a $T_{data}$ transaction, it is sent to the blockchain which is able to route a pointer to data on public ledger (the pointer is SHA-256 hash of data). The user and service can query the data by using a $T_{data}$ transaction in association with a pointer (key). Then the verification of digital signature by blockchain is done for knowing either it belongs to a user or a service. If it belongs to a service, then the permissions for accessing the data are also checked. $T_{access}$ transaction can be issued anytime for changing the permissions granted to a service at any time that includes revoking access to previously stored data. It is possible to overview one's data and ability to alter permissions by the development of a web-based dashboard and it is similar as developing centralized wallets such as Coinbase for Bitcoin1. The implementation of Kademilia[13] stores the key value of off-blockchain or a distributed hashtable (DHT). A network of nodes is used to maintain DHT for fulfilling approved read/write transactions. The high availability is ensured by sufficiently randomizing data across the nodes. For storing, the alternative solutions to off-blockchain should be considered. For storing the data, a centralized cloud could be used. It offers advantages in ease of deployment and scalability as well as some degree of trust in third party.

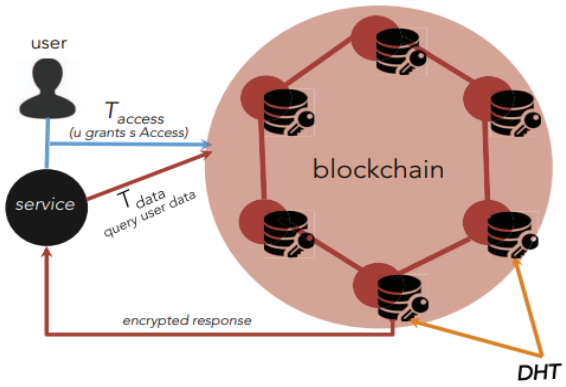


**Fig. 10 Summary of decentralized platform**

## IV. THE NETWORK PROTOCOL

The protocol used in the system is described here. Our platform uses the standard cryptographic building blocks: the 3-tuple defines a symmetric encryption scheme(Genc, Eenc, Denc) which is known as "generator, encryption and decryption algorithms" respectively, a "digital signature scheme" (DSS) which is also described by 3 tuple namely, (Gsig, Ssig, Vsig) namely, "generator, signature and verification algorithms" respectively which are implemented by SHA-256.

### *A. Building Blocks:*

The bitcoin and blockchains are earlier discussed. 1) Identities: A pseudo identity mechanism is utilized by blockchains. The privacy can be increased by using as many public-keys as he wants for generating many pseudo identities as desirable. The extension of compound identities will be used in our system. A shared identity for two or more parties is known as a compound identity, where at least one party owns the identity and the rest are restricted as guests as they have restricted access. The implementation of user and the service is illustrated by protocol 1. The signing key-pairs of owner and guest are comprised in the identity also the symmetric key for encryption and decryption of data for protecting data from any threat in the system. A compound identity when seen by a network is the 2 tuple:

$$Compound_{u,s}^{(public)} = (pk_{sig}^{u,s}, pk_{sig}^{s,u}) \quad (1)$$

A 5 tuple is followed is by the entire identity which includes the private keys:

$$Compound_{u,s} = (pk_{sig}^{u,s}, sk_{sig}^{u,s}, pk_{sig}^{s,u}, sk_{sig}^{s,u}, sk_{enc}^{u,s}) \quad (2)$$

**Protocol 1** Generating a compound identity

1: **procedure** COMPOUNDIDENTITY$(u, s)$
2: $u$ *and* $s$ *form a secure channel*
3: $u$ executes:
4: $(pk_{sig}^{u,s}, sk_{sig}^{u,s}) \leftarrow \mathcal{G}_{sig}()$
5: $sk_{enc}^{u,s} \leftarrow \mathcal{G}_{enc}()$
6: $u$ *shares* $sk_{enc}^{u,s}, pk_{sig}^{u,s}$ *with* $s$
7: $s$ executes:
8: $(pk_{sig}^{s,u}, sk_{sig}^{s,u}) \leftarrow \mathcal{G}_{sig}()$
9: $s$ *shares* $pk_{sig}^{s,u}$ *with* $s$
10: // Both $u$ and $s$ have $sk_{enc}^{u,s}, pk_{sig}^{u,s}, pk_{sig}^{s,u}$
11: **return** $pk_{sig}^{u,s}, pk_{sig}^{s,u}, sk_{enc}^{u,s}$
12: **end procedure**

2) Blockchain memory: L is the blockchain memory space, represented as hastable L: $\{0,1\}^{256}$ to $\{0,1\}^{N}$ for N>> 256 and it is capable of storing sufficiently large documents. The adversarial models used in bitcoin and other

blockchains assume this memory to be tamperproof. The following simplified and albeit inefficient implementation can be considered for implementing trusted data store on a blockchain. The serialized output is constructed by rest of the outputs. Along with insertion, updation and deletion operations are also allowed because only recent most transactions are returned while looking up L[k].

3) Policy: a set of permissions granted by users is denoted by POLICY u.s. for example, on installing a mobile application that needs to access the location and contacts of a user, POLICY u.s.={location, contacts}. Considering that protocol will not be subverted and data will not be labelled incorrectly, it is possible to store any data safely.

4) Auxiliary Functions: The message sent to transaction, containing the arguments is de serialized by P arse(x). The verification of appropriate permissions to originator is illustrated by protocol 2 by CheckPolicy(pkk sig,xp).

**Protocol 2** Permissions check against the blockchain

**procedure** CHECKPOLICY($pk^k_{sig}, x_p$)
  $s \leftarrow 0$
  $a_{policy} = \mathcal{H}(pk^k_{sig})$
  **if** $L[a_{policy}] \neq \emptyset$ **then**
    $pk^{u,s}_{sig}, pk^{s,u}_{sig}, POLICY_{u,s} \leftarrow Parse(L[a_{policy}])$
    **if** $pk^k_{sig} = pk^{u,s}_{sig}$ **or**
($pk^k_{sig} = pk^{s,u}_{sig}$ **and** $x_p \in POLICY_{u,s}$) **then**
      $s \leftarrow 1$
    **end if**
  **end if**
  **return** $s$
**end procedure**

### B. *Blockchain Protocols:*

The core protocols executed on a blockchain are described in blockchain protocols. After the reception of $T_{access}$ transaction, node 3 in a network executes protocol 3 whereas $T_{data}$ transactions executes protocol 4. A POLICYu.s. Set is sent to users for changing the set of permissions granted to guest by $T_{access}$ transactions. All access rights that were previously granted are revoked by sending an empty set. A user sends $T_{access}$ transaction compound identity for signing up to a service for the first time. The read/write operations are governed by $T_{data}$ transactions. CheckPolicy enables only the user or a service to access the data.

### C. *Privacy and Security Analysis:*

It is desirable that blockchain should be tamper-free because it needs a wide variety of untrusted peers. A secure manner is required for managing keys by a user. Consider an example of a 'secure-centralized wallet service'.

**Protocol 3** Access Control Protocol

**procedure** HANDLEACCESSTX($pk^k_{sig}, m$)
  $s \leftarrow 0$
  $pk^{u,s}_{sig}, pk^{s,u}_{sig}, POLICY_{u,s} = Parse(m)$
  **if** $pk^k_{sig} = pk^{u,s}_{sig}$ **then**
    $L[\mathcal{H}(pk^k_{sig})] = m$
    $s \leftarrow 1$
  **end if**
  **return** $s$
**end procedure**

**Protocol 4** Storing or Loading Data

**procedure** HANDLEDATATX($pk^k_{sig}, m$)
  $c, x_p, rw = Parse(m)$
  **if** $CheckPolicy(pk^k_{sig}, x_p) =$ `True` **then**
    $pk^{u,s}_{sig}, pk^{s,u}_{sig}, POLICY_{u,s} \leftarrow Parse(L[\mathcal{H}(pk^{u,s}_{sig})])$
    $a_{x_p} = \mathcal{H}(pk^{u,s}_{sig} \parallel x_p)$
    **if** $rw = 0$ **then** ▷ rw=0 for write, 1 for read
      $h_c = \mathcal{H}(c)$
      $L[a_{x_p}] \leftarrow L[a_{x_p}] \cup h_c$
      (DHT) $ds[h_c] \leftarrow c$
      **return** $h_c$
    **else if** $c \in L[a_{x_p}]$ **then**
      (DHT) **return** $ds[h_c]$
    **end if**
  **end if**
  **return** $\emptyset$
**end procedure**

By using this protocol, it is possible for a user to control their data. The adversary could not pretend as a user or forge a digital signature for corrupting the network or controlling the majority of network resources. It is not possible for an adversary that controls DHT nodes to learn raw data because it has encryption with keys that are not possessed by any of the nodes. It is possible to obtain signing and encryption keys when a new compound identity is generated. The data is assumed to be safe if only one of the keys is obtained by adversary. The exposure of a single compromised identity could be limited by splitting the identities. For every hundred records stored, it is possible to generate new key.

## V. DISCUSSION OF FUTURE EXTENSION

The present future extensions to blockchains is presented in this section. In comparison to existing art systems, a better trusted computing platforms is shaped in this way. This is able to upsurge the utility of earlier presented platform.

### A. *From storage to processing:*

Any function [14] could be securely evaluated by using Multiparty Computation

(MPC) by splitting data into shares, instead of encrypting it. Shamir's Secret Sharing is implemented. Figure 2 demonstrates the working of blockchains specifically in our framework. Results are safeguarded against tampering by storing them on a public ledger. Thus, the results of elections can be seen but nobody is able to learn about whom the individual voted for.

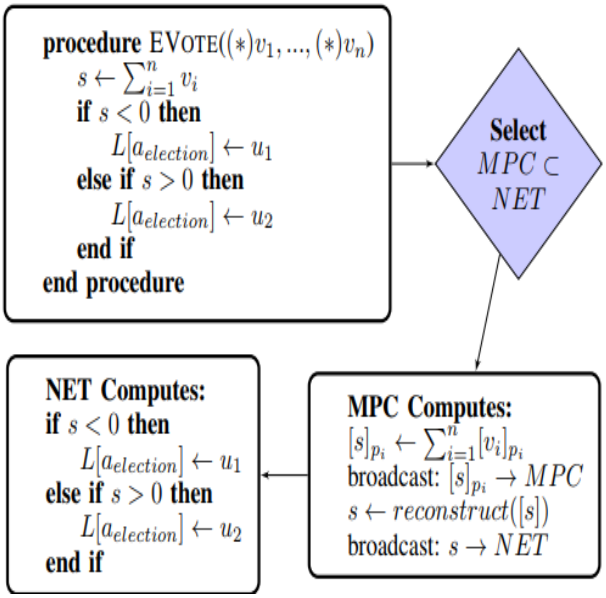


**Fig. 2 Secure communication in blockchain**

*B. Trust and Decision Making in Blockchains:*

the proof of work algorithm [15] states that computational resources are solely responsible for collective decision making process of a bitcoin and all the nodes are equally untrusted. Also it means that for a "node n, trustn ∝ resources (n) (probabilistically)". The weight of a node in votes is decided by it. Proof of work algorithm also states that there are lesser chances of cheating by a node. On the basis of node behaviour, rewards are given to good actors that follow the protocol. The probability p is the expected value as we are using a binary random variable. This probability is approximated by counting the number of bad and good actions taken by a node and then squashing it into a probability by using a sigmoid function. The every score had a trust score re-evaluated in every block *i* as:

$$trust_n^{(i)} = \frac{1}{1 + e^{-\alpha(\#good - \#bad)}}, \quad (3)$$

Where α denotes the step size. More weight to trusted nodes is given by the network and blocks are computed more efficiently. The system should be able to resist Sybil attacks because trust within the system is earned gradually. It is possible that other types of attacks are attracted by this system, such as increase of reputation of nodes for acting maliciously at a later time. It is mitigated by random selection of nodes, that are weighted by their trust for voting on each block and then equally weighted majority vote is taken. Regardless of the trust level of actors, too much influence of single actors is prevented by this.

## VI. CONCLUSION

Third parties should not be trusted for personal and sensitive data because they can be misused and attacked. Instead of this users should themselves control as well as own data without negotiating with the ability of authorities for providing personalized services. Trusting any third party is not required by the users and they are aware of the use of data that is collected about them. The users are recognized as the owners of their personal data by the blockchain. The companies utilize data without taking care of proper security. Also, the decisions about storing, sharing and collecting data with a decentralized platform should be made simpler. For enforcing the laws and regulations automatically, they should be programmed into a blockchain itself. The data is accessed or stored by making a ledger as legal evidence because its computation is tamper-proof..

## REFERENCES


1. ScienceDaily, "Big Data, for better or worse: 90% of world's data generated over last two years," 2013.
2. P. Vagata and K. Wilfong, "Scaling the Facebook data warehouse to 300 PB," Official Facebook Code Engineering Blog, 2014. .
3. M. Lesk, "The Seven Ages of Information Retrieval," Int. Fed. Libr. Assoc. Institutions, 1995.
4. K. Schwab, A. Marcus, J. R. Oyola, W. Hoffmann, and M. Luzi, "Personal Data: The Emergence of a New Asset Class," 2011.
5. V. Goel, "Facebook Tinkers With Users ' Emotions in News Feed Experiment , Stirring Outcry," The New York Times, 2014.
6. J. Ball, "NSA's Prism surveillance program: how it works and what it can do | World news | guardian.co.uk," Guard., 2013.
7. H. Nissenbaum, "Respect for context as a benchmark for privacy online: What it is and isn't," in Social Dimensions of Privacy: Interdisciplinary Perspectives, 2015.
8. L. SWEENEY, "k-ANONYMITY: A MODEL FOR PROTECTING PRIVACY," Int. J. Uncertainty, Fuzziness Knowledge-Based Syst., 2002.
9. A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "Diversity : Privacy Beyond k-Anonymity," ACM Trans. Knowl. Discov. Data, 2007.
10. N. Li, T. Li, and S. Venkatasubramanian, "Closeness: A new privacy measure for data publishing," IEEE Trans. Knowl. Data Eng., 2010.
11. G. Barthe, B. Köpf, F. Olmedo, and S. Zanella-Béguelin, "Probabilistic Relational Reasoning for Differential Privacy," ACM Trans. Program. Lang. Syst., 2013.
12. C. Gentry, "Fully homomorphic encryption using ideal lattices," in Proceedings of the 41st annual ACM symposium on Symposium on theory of computing - STOC '09, 2009.
13. P. Maymounkov and D. Mazières, "Kademlia: A Peer-to-Peer Information System Based on the XOR Metric," 2002.
14. M. Ben-Or and A. Wigderson, "Completeness theorems for non-cryptographic fault-tolerant distributed computation," 2003.
15. S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System(HP)," Consulted, 2008.