ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/325966342

Practical Deployability of Permissioned Blockchains

Conference Paper · July 2018

CITATIONS
0

READS
242

5 authors, including:

Nitesh Emmadi
Tata Consultancy Services Limited
6 PUBLICATIONS **17** CITATIONS
SEE PROFILE

Harika Narumanchi
Tata Consultancy Services Limited
6 PUBLICATIONS **17** CITATIONS
SEE PROFILE

Some of the authors of this publication are also working on these related projects:

NTRU Cryptosystem View project

Fully Homomorphic Encryption (FHE) View project

All content following this page was uploaded by Harika Narumanchi on 06 July 2018.

The user has requested enhancement of the downloaded file.

Practical deployability of Permissioned Blockchains

Nitesh Emmadi[1], Vigneswaran R[2], Srujana Kanchanapalli[3],
Lakshmipadmaja Maddali[4], and Harika Narumanchi[5]

TCS Innovation Labs, INDIA
[1]nitesh.emmadi1@tcs.com,[2]vigneswaran.r@tcs.com,
[3] srujana.k@tcs.com,[4] lakshmipadmaja.maddali@tcs.com,
[5] h.narumanchi@tcs.com

Abstract. Ever since the evolution of cryptocurrencies, there has been profound interest in employing the underlying Blockchain technology for enterprise applications. Enterprises are keen on embracing the advantages of Blockchain in applications ranging from FinTech, Supply chain, IoT, Identity Management, Notary, Insurance and to many other domains. Blockchain is often spoken of as the third disruption after computers and the internet, and is being studied for application in several domains. A blockchain, as used in most cryptocurrencies, does not require any authorization for participants to join or leave the system, and hence is referred to as a *permission-less* blockchain. However, enterprise applications cannot operate in such models. Enterprise applications operate in a regulated, *permissioned* blockchain setting. This paper provides an industry focused insight into the practicality and feasibility of permissioned blockchains in real-world applications. In particular, we consolidate some non-trivial challenges that should be addressed in making the *permissioned* blockchain practically deployable in enterprises.

Keywords: Permissioned blockchain· challenges · smart contracts · enterprises

1 Introduction

It is always desirable to have a system which is not controlled by a single entity. A system that is governed by multiple entities is inherently more trusted in the sense that a single malicious entity in charge cannot manipulate the system. For instance, the cryptocurrency Bitcoin [1] is one such system that functions without a central controlling authority. Rather, it is controlled collectively (in agreement) by the parties involved in the system, thereby distributing the trust. Bitcoin enables monetary transactions without the need for a central authority like a bank. Such systems are referred to as having decentralized trust i.e., systems where trust does not depend on any one specific entity. It is this feature that drew the attention of academia and industry.

The core primitive that enables decentralization in Bitcoin is blockchain. Blockchain can simply be viewed as a shared append-only database with entries (transactions) agreed upon by all the involved parties. The entries are grouped into a *block* and appended to the ledger. A typical blockchain as in Bitcoin is referred to as *permission-less* blockchain. It means that entities can join/leave the blockchain network without any restrictions. The identities of these entities are not verified i.e., the entities are anonymous. However, enterprise applications cannot do the same. Enterprises need the entities to have verified identities in order to enable accountability and traceability. Hence, they need a permissioned model of blockchain or *permissioned* blockchain.

Another important difference between *permission-less* and *permissioned* blockchain is the consensus (agreement) mechanism. *Permission-less* blockchains are suitable for trust-less environments without accountability due to anonymity. They often employ a computationally intensive mechanism, usually *proof of work* [5] for consensus. Proof of work is a self-rewarding computational challenge for block formation, to be solved by any of the entities in the network. The entity that solves the puzzle propagates the solution along

with the block to other parties in the network. Acceptance of the propagated block by other peers establishes correctness of the solution and block formation. Moreover, computational complexity of the proof of work guarantees that it is in-feasible to tamper with the ledger without significant computational effort. In case of *permissioned* blockchains, the entities in the system have verified identities. Hence, consensus can be reached simply through non-repudiating interactions. This study focuses only on *permissioned* blockchains due to their applicability in enterprise environments.

The paper is organized as follows: In Section 2 we give a brief overview of blockchain. Section 3 outlines the blockchain in a permissioned environment. In Section 4 we demonstrate the applicability of blockchain in different usecases. Section 5 we describe how blockchain adds value to commonly occurring usecases and in Section 6 we emphasize on the challenges in making the blockchain practically deployable.

2 Brief Overview of Blockchain

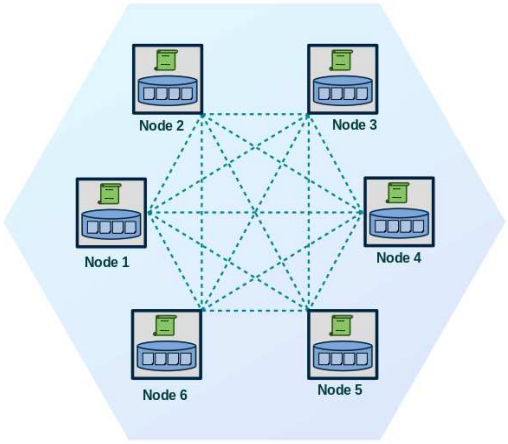

Fig. 1. Blockchain Network

Blockchain is a shared ledger that enables mutually distrusting parties to transact with each other without any central authority. The participants together form a peer-to-peer network of nodes with a common ledger (Figure 1). Blockchain, as its name implies, is a *chain of blocks*. Each block consists of a set of entries (transactions in case of cryptocurrencies) to be included in the blockchain and each new block is chained to the preceding block. All entries are appended to the ledger based on the consensus (agreement) of the involved parties. This ensures that the ledger is always consistent among all the parties.

2.1 Chaining the Blocks: Hashchain

The chaining in blockchain is done by a special cryptographic primitive called *hash function*. A hash function is a one-way function that takes any arbitrary string as input and easily computes another string of pre-determined length. Important properties of a hash function are:

- Given the output, it should be computationally in-feasible to find input
- Highly sensitive to changes in input, that is, a single bit change in input should affect several bits of the output

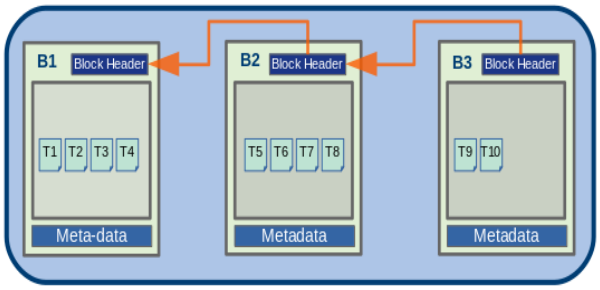

Fig. 2. Blockchain

– Computationally in-feasible to change the message without changing the hash
– Computationally in-feasible to find two messages with the same hash

In blockchain, the blocks are chained together by including the hash of a block in the next block (Figure 2). This method is called *hash-chaining*. This way of chaining in blockchain makes it immutable, that is, for a continuous chain of blocks, if modification is done on a block "n" and a new hash is produced, then all the blocks after "n" will produce hashes that are different from the previously known hashes. The hash function ensures integrity and provides tamper evidence of the blockchain. Also, the fact that the ledger is shared among the parties ensures that one malicious party cannot alter the ledger without the knowledge of others.

2.2 Consensus

Consensus is an agreement among the parties involved in the blockchain network. All the data in the blockchain is validated by the participants before being written into the ledger. Consensus ensures consistency of the ledger across all parties. Permission-less systems operate in trust-less environments and thus rely on consensus mechanisms like proof of work [5] or proof of stake [2] or proof of space [3] and so on. *Permissioned* blockchains rely on consensus mechanisms like Practical Byzantine Fault Tolerance (PBFT) [7], SIEVE [8] and so on.

2.3 Smart Contracts

A smart contract is an encapsulation of business logic. Applications have several computations and validations before writing data on the ledger. Smart contracts enable embedding this logic to operate on the data. Smart contracts handle reading or writing data on the blockchain. Smart contracts are executed by one or more parties independently to ensure the correctness of data processing, and to impart trust to the system. The results of the smart contract execution from various parties is used for consensus purposes.

2.4 Properties of a Blockchain

The following are some interesting properties of blockchain that are applicable to both permission-less and permissioned environments:

– **Decentralization:** Blockchain enables mutually distrusting parties to engage with each other without a central authority.
– **Transparency:** Shared ledger ensures all the transactions are visible to all involved parties.

– **Integrity:** Immutable ledger preserves the integrity of the data. Shared ledger obtained through consensus ensures tamper detection.
– **Availability:** Shared ledger ensures there is no single point of failure.

Blockchain is being studied for application in enterprise use-cases for one or more of the above advantages.

3 Blockchain for enterprises

A *permissioned* blockchain incorporates several features which are not present in *permission-less* blockchains. These features are necessary to ensure applicability of blockchains to enterprises. Figure3 illustrates the application flow of *permissioned* blockchain. Briefly, the application flow of a typical *permissioned* blockchain has a client application SDK that submits a transaction proposal to the peer. Once consensus is achieved, each peer updates their respective copy of the blockchain ledger. The stateDB holds the current state of all the entities (users). Finally Peers notifies the client applications that the ledger has been updated. Note that we do not discuss any particular instantiation of *permissioned* blockchain rather we focus on *permissioned* blockchain in general. Some of these features include:

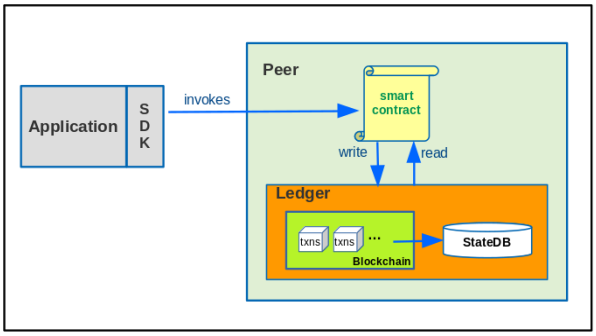

Fig. 3. Application Flow of *Permissioned* Blockchain

3.1 Privacy and Security

Blockchain is a shared ledger visible to all the involved parties. However, enterprise applications are concerned about security and privacy of the customers' data. For example, two contending businesses availing services on the same blockchain do not wish their data to be available to each other.

The privacy and security requirements in *permissioned* blockchain can be listed as follows:

– **Confidentiality:** A transaction on blockchain should not be accessed by an unauthorized party.
– **Unlinkability:** Different transactions of the same user should not be linked to each other by an unauthorized party.
– **Anonymity:** Any transaction recorded on the blockchain should not be associated with the user by an unauthorized party.

Cryptographic primitives ensure confidentiality, unlinkability and anonymity in blockchain.

3.2 Attribute based access controls

Attribute based access controls is a security mechanism that provides access based on the attributes of the requester. Most enterprise applications are modeled with attribute based access controls for flexibility and granularity. This is necessary in order to define access privileges based on attributes of the individuals accessing an application, rather than on the individuals themselves. *Permissioned* blockchain platforms should provide attribute based access controls in order to enable enterprises to realize the applications with granular access controls. In the *permissioned* blockchains, these access control mechanisms are embedded into smart contracts 2.3.

3.3 Auditability

Permission-less systems, like cryptocurrencies, are not regulated and cannot be audited as the involved parties are anonymous. Enterprise applications have to comply with audit requirements by both internal and external authorized parties. The auditor may need to access the blockchain ledger partially or fully, either temporarily or on a continuous basis. Enterprise blockchains should provide a mechanism to facilitate such audits. Some platforms often encrypt the data with derived keys in a hierarchical fashion to enable providing access to only specific portion of the data. An auditor should also be able to associate the pseudonymous identities on the blockchain with the actual identities.

4 Do you need a blockchain?

The advantages of blockchain are being studied for several applications ranging from IoT to international monetary transfers and more. Blockchain has the potential to provide significant value addition to some but not all applications. It is easy to get carried away by the hype around blockchain. Hence, a critical evaluation of the applicability of blockchain in use-cases is needed.

Fig 4[1] provides guidelines for evaluating applicability of blockchain for use cases. The original flowchart from [4], assumes the existence of an always-online trusted third party (TTP) for certain use-cases. However, there is no such thing in the real world. Another important modification is at the "multiple writers" flow. Blockchain is also being considered for applications involving a single writer, for example Land Records Registry or Motor Vehicles Registry or Logging (System Logs/Event Logs). These category of use-cases depend on blockchain for long-term integrity purposes. This kind of use-cases are valid only if the data entry to the blockchain is trusted.

Note that Figure 4 classifies *permissioned* blockchains into *Public Permissioned* blockchain and *Private Permissioned* blockchain. The distinction arises from the public verifiability of blockchain data. Public Permissioned blockchain can be publicly verified by anyone whereas in a *Private Permissioned*, it is not allowed.

5 Use-cases

In this section, we present several enterprise use-cases and explain how blockchain adds value in each case. We only consider how blockchain improves the operational efficiency of the usecases rather than focusing on design choices.

[1] The modifications are based on independent comments from Prof. Bart Preneel, University of Leuven, Belgium at BLOCKCHAIN 2017 Workshop and Sitaram Chamarty, TCS Innovation Labs, Hyderabad.

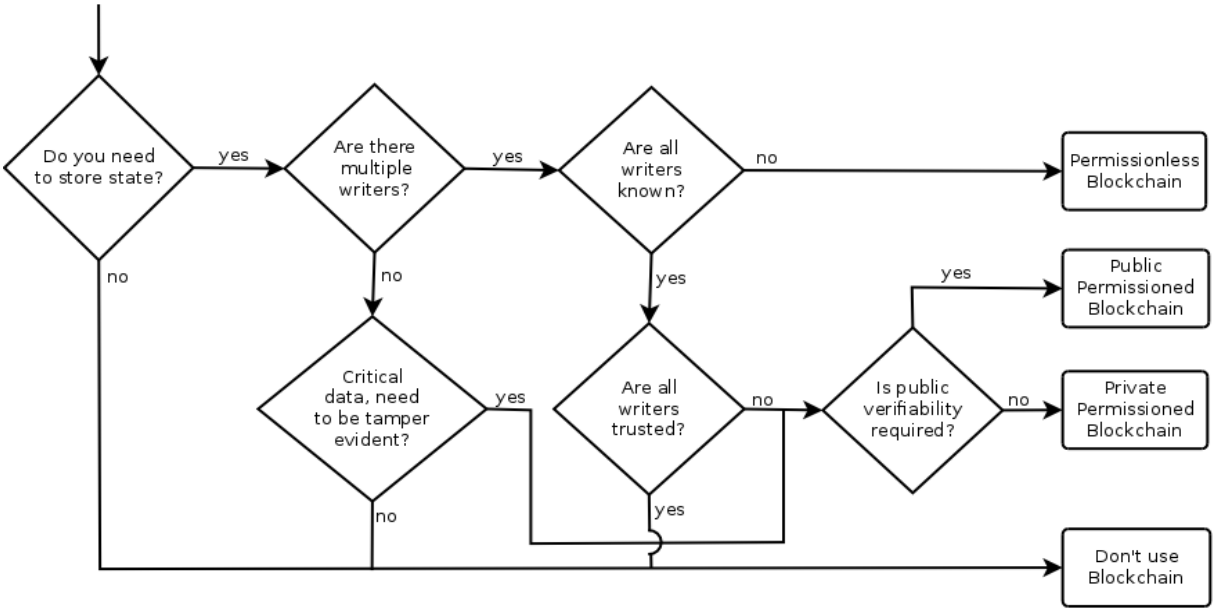

Fig. 4. Do you need a blockchain? (This is a modified version of flowchart from [4])

5.1 Global Payments

International transfers in banking is often a tiring process. Consider an international transfer request submitted to a local bank. The local bank could not process the payment directly as it does not deal with foreign exchange. The local bank requests foreign exchange from a Forex bank that deals with foreign exchange. This foreign exchange then has to be routed to the receiver through another routing bank. In the current banking system, this settlement is done with several requests among the banks and can take significant effort.

Blockchain can be used to speed up the process of payments across multiple banks. Blockchain can bring all the banks into a single network and transactions can be made easily. All the requests for money transfer done between banks are recorded in the blockchain. So, it reaches all the parties involved in the blockchain system. This makes the reconciliation process instantaneous and improves the speed of the payments.

5.2 Delivery versus Payment (DVP)

Delivery versus payment, or DVP, is a common form of settlement for securities. The process may involve transfers of two securities in such a way as to ensure that delivery of one security occurs if and only if the corresponding delivery of the other security occurs. This is done to avoid settlement risk where one party fails to deliver the security when the other party has already delivered the payment. The parties in DVP use-case are buyer, seller, clearing house, bank and securities issuer. In the usual scenario, the process of settlement is done by the parties after ensuring that securities are transferred. This induces delay in the settlement process.

With blockchain, the settlement can be reached atomically on the shared ledger. For a securities exchange, both buyer and seller submit transactions to the blockchain. The transaction submitted by the seller contains a message to the corresponding asset issuer to transfer a particular asset to the buyer, and a message to the DVP provider to receive corresponding payment in return. Similarly, the transaction submitted by the buyer includes a message to the bank to transfer payment to the seller, and a message to the DVP provider

to receive the asset in return. The DVP provider makes sure that both occurs in an atomic way. The buyer's and seller's transaction also includes post trade settlement instructions to the Clearing house. This triggers the Clearing house to submit transactions to both the bank and the security issuer to get the settlement done. Blockchain improves the operational efficiency by providing atomic on-chain settlements.

5.3 Digital Identity (eKYC)

Enterprises need to validate identity of their customers before providing services. This is essential for regulatory and compliance purposes. The process of identifying and verifying identity of clients is generally referred to as "Know Your Customer (KYC)". It is redundant and often expensive for all the organizations to verify their customers independently. It is also an inconvenience for customers to provide the same information to different organizations at different times.

It can be seen that a blockchain can enable data sharing between multiple organizations and eliminate redundant verification. If one organization verifies KYC of a user and records it in a blockchain, the other organizations can access the information from the blockchain without independently repeating the KYC process time and again. This eliminates unnecessary expenses for the organizations and improves the user experience while also recording the details of the KYC in a tamper-evident database. Blockchain, in this case, improves operational efficiency through data sharing.

5.4 Receivables Registry

Receivables registry is a peripheral process for Trade Receivables Discounting System (TReDS). Receivable registry maintains financing invoices in TReDS. The invoices are registered into the system by TReDS operators or banks. The invoices can be queried by other financiers in the system. It is possible for parties to register the same invoice fraudulently for multiple financing.

Duplication of invoices can be solved by leveraging blockchain among the parties. With blockchain, a duplicate entry can easily be detected by validating against the shared ledger. Different parties can have the full copy of the same receivables registry and process the invoice independently without worrying about duplicate entries being registered. Blockchain enables different parties to register, update, search and query invoices on the receivables registry, serving to prevent fraud by efficiently detecting duplicates.

5.5 Supplychain

Supplychain tracks the life cycle of a product from manufacturing to purchase by the end user. Blockchain is believed to add immense value to the supply chain ecosystem. A typical supplychain management system requires maintaining records for products, tracking shipments and notifying related parties about the status. Transparency in such workflows enhances the operational efficiency as the updates are visible to all the involved parties. Blockchain enables transparency in the flow. In turn, this helps identify counterfeit products entering into the supply chain midway (not introduced at the origin itself). Also, the effectiveness greatly depends on the type of product. For example, one way of eliminating chip counterfeits is to capture the location where the microchips were created and process the specific port of entry [20]. If there is a mismatch in the port of entry, the microchips would be held by customs and a notification is sent to the appropriate parties detailing the counterfeits. Blockchain also improves inventory management by providing real-time visibility.

Moreover, blockchain can also be leveraged for recording the provenance of high value assets like diamonds [16]. Blockchain can help track diamonds by creating a permanent record of the asset's history and ownership. Blockchain ledger provides proof of authenticity for the asset.

5.6 Internet of Things

Internet of Things (IoT) is a network of physical devices that exchange data streams. IoT usually consists of low-powered and low-storage devices like sensors. These suffer from resource constraints and can only perform limited tasks. Blockchain, on the other hand, deals with huge amount of storage(and possibly computations). When blockchain and IoT are imagined together it is usually the IoT infrastructure outside the sensors that is imagined in the blockchain scenario.

Blockchain is being studied in IoT use-cases to deal with access controls, data storage, auditability, decentralized (autonomous) networks, and so on [11]. However, the problems outside the blockchain system in IoT still exists and can devalue the blockchain advantages. Extensive study and further research into the practical aspects is needed.

5.7 Travel Ecosystem

Airlines travel ecosystem consists of travel providers, Global Distribution System (GDS), and travel buyers. Travel providers are service providers such as airlines, hotels, and so on and the travel buyers include consumers such as travel agencies or individual travelers. GDS is a network of travel providers and travel buyers. Air ticket or hotel booking can be done directly through the airline/hotel's website. However, that direct mode of booking is a very small proportion. More than 60% of bookings are indirect, through travel agencies or third parties. GDS plays a predominant role in aggregating the seat availability related information from various airlines. A recent study has shown that three GDSs dominates the market with 99% of market share [10]. Though GDS is trusted, there is a possibility of monopoly given that the GDS has significant negotiating power. Moreover, due to the complex pricing rules, it is hard to verify that the offers provided by travel providers are genuine.

There is a need for a trusted intermediary to enable airline search, multiple airline combinations search, visibility for providers to enable them to provide better services. A blockchain can be leveraged as an intermediary (with or without replacing GDS) to enable data sharing and transparency. Blockchain can provide a transparent trustworthy network and prevent disputes in the system.

6 Challenges

Blockchain in the current state is not practically deployable for almost all use-cases. In this section, we highlight several challenges in *permissioned* blockchains.These challenges are generic and prevail in all the usecases.

6.1 Throughput

All the current enterprise applications rely on centralized databases to store data. Centralized databases are very efficient and are capable of handling a high number of read/write operations without much latency. However, blockchains are decentralized and hence inherently slower due to consensus. The throughput issue has two aspects.

- implementation related: choice of various options (for example, "execute-order-validate" vs "order-execute" paradigms) and tunables such as payload size, batch size, block latency and so on [19].
- intrinsic: because by definition, what was previously the equivalent of one simple database *insert* or *update* is now a much more complex process that involves several messages going back and forth between several nodes, and so on.

This makes application of blockchain more ideal for low throughput use cases dealing with high value (or at least notional value) transactions. There is a need to improve throughput of blockchain to accommodate it in wider applications.

6.2 Consensus

Consensus mechanisms are complex and inefficient due to several factors like message complexity, communication rounds between nodes, etc. This severely impacts the throughput. Significant industry and academic research is concentrated on improving efficiency. Alternate solutions, such as trusted computing, are being evaluated as replacement in this regard. Also, a single consensus algorithm that suits all types of blockchain use cases is almost impossible to achieve. Thus there is a need for pluggable consensus framework with parameter tuning such as message/computational latency. Designing such a framework with automatic consensus algorithm suggestions is a challenge.

6.3 Query efficiency

As per the confidentiality requirement, data in the blockchain is encrypted, thus making it harder to query efficiently. There is a need to investigate cryptographic primitives like searchable encryption in order to solve this problem. The usual searchable encryption schemes allow queries on data encrypted under single key. However, the data on blockchain is encrypted under several different one-time keys. Searchable encryption schemes that can handle such requirement are necessary to improve query efficiency.

6.4 Privacy & Confidentiality

Achieving privacy and confidentiality in a shared ledger is a challenge. The current solutions depend on standard cryptographic primitives to achieve privacy and security. However, these solutions are either inefficient or do not quite solve the problem. Though new paradigms like Zero Knowledge Proofs (ZKPs) for developing anonymous credentials (Identity Mixer [9]) or privacy-preserving smart contracts [12] are being explored in this regard, there are still open issues. For instance, cryptographic algorithms and their implementations demand more scrutiny before actually being deployable in the real world. Dependence on such novel and not yet widely deployed cryptographic primitives induce risk in dealing with customers' privacy and confidentiality.

6.5 Key Management

Usability in security has been a challenge for ages. Most current systems depend on passwords for protection. Though the password is the most well-known security mechanism, it still suffers from lack of "non-repudiation". Blockchain systems, therefore, require users to use asymmetric cryptographic keys. Storing the privatekeys is an unmanageable risk with current levels of user education. Though several secure wallets for key storage are developed, they are often exploited. The immutable nature of data on the ledger adds several more challenges in key rotation and key loss.

6.6 Inter-blockchain communication

Enterprise applications often involve multiple systems communicating with each other. To cater to this kind of application, use-cases are now envisioned in a multi-blockchain model, which is basically a collection of blockchains. Establishing cross blockchain communication without compromising integrity guarantees, privacy and confidentiality is a big challenge. There is a need to develop proper cryptographic mechanisms to ensure privacy in this setup.

Moreover, applications developed on different *permissioned* blockchain platforms should be able to communicate with each other. Standard communication protocols and mechanisms must be supported uniformly by all the blockchain platforms.

6.7 Auditability

Auditability allows authorized entities(internal/external) to audit transaction records. To perform an audit, auditors should be given a means to investigate the activity of users or their transactions. The current *permissioned* blockchains envision pseudonymous identities and encrypted transactions for privacy and confidentiality. Auditability requires a provision to de-anonymize the users and link their pseudonymous identity to the original identity at the time of enrollment and also de-classify encrypted transaction data. This can be done by providing necessary secret keys to the auditors. However, the accessibility of blockchain endangers the privacy and confidentiality of users if the secret keys are shared to other parties. Alternatively, access control mechanisms can be leveraged to avoid sharing the keys but this introduces the additional requirement of proving the correctness of the data accessed. Auditability still remains one of the important requirement to be fulfilled. Hence, there is a need to develop comprehensive mechanism for dealing with the audit management. Zero-Knowledge Proofs can be potential solution to present proofs of correctness to the auditors.

6.8 Compliance with the privacy regulations

Compliance with privacy regulations can be more challenging in the blockchain world. For instance, General Data Protection Regulation (GDPR) mandates a "Right to Forget" feature for customers. The immutable nature of blockchain is contrary to this requirement. There are mutable blockchain [18] solutions which can cater to these needs, however they call into question the very basis of what makes a blockchain attractive for given use. Thus, more research is needed to make blockchains compliant with regulations.

6.9 Migration

Migration of data from existing applications to new blockchain systems can be a problem. The data models for blockchain application can differ from the existing systems. The data migration has to be seamless and ensure services without disruption.

6.10 Interoperability with existing systems

One of the crucial challenges in developing blockchain systems for existing applications is interoperability of blockchain with existing components in the applications. Rather than creating entirely new applications, it is better to remodel existing applications at certain levels to embed the blockchain. The blockchain system should be able to integrate itself with the other existing components in the application like query engines, front-end systems, UI systems and etc. This makes it necessary to structure new data models so as to retain sufficient query capability to support existing front-end systems and UIs.

6.11 Smart Contract security

In recent times, several vulnerabilities in various smart contracts have been exploited in the blockchain platforms like Ethereum[6]. The immutability of blockchain makes dealing with the after-math very challenging. Security bugs in applications are inevitable. Therefore some notions of resilience need to be explored in order to avoid irreversible damages.

6.12 Standardization

Standardization ensures systems or services to better realize thier objectives in secure and robust manner. Standardization of blockchain is important in realizing real-word applications. Blockchain standardization enables interoperability, compatibility, security, regulatory compliance. Standards also encourage innovation and embed more confidence in building applications.

6.13 Oracles

Often the smart contracts in a blockchain are required to validate transactions based on information from external systems. Some smart contracts may have to be triggered based on information from the physical world. To a blockchain, such information is delivered by external entities referred to as oracles. An oracle acts as an interface between the blockchain and external systems. Relying on oracles for such information is a necessary risk. There is a need to mitigate the risk by ensuring the authenticity of information transmitted to the blockchain. Another important challenge is confidentiality and privacy of oracle queries. Some solutions like Town Crier [14], Oraclize [15] explore oracles in context of decentralized applications.

6.14 Certificate Authority and Revocation Lists

The current *permissioned* blockchains rely on traditional Public key infrastructure (PKI) to manage identities of the entities. Even though the blockchain is decentralized, this certificate issuing system for issuing the certificates is based on a central authority. This could lead to issuance of spurious certificates to unauthorized parties. Therefore, the PKI infrastructure used in blockchain must be made decentralized. Even if a malicious entity obtains a certificate from the CA it can be easily detected by other nodes. There are solutions such as [13,17] that provide transparency and prevent misbehavior however further study is necessary to incorporate such solutions into *permissioned* blockchains. Similarly, it is also necessary to have a decentralized solution for certificate revocation (CRL or OCSP).

7 Conclusion

A comprehensive elucidation of current industrial experience with blockchains have been provided in this paper. It is necessary for market players to understand and evaluate the practical aspects of blockchain applications before deploying blockchain based applications. Our study enumerates several challenges and limitations in current blockchain models that need to improved before the full potential of blockchain can be realized. Blockchain is believed to be the third disruption, however, our understanding suggests that it is **"not yet"**.

8 Acknowledgements

We thank Dr. Sumanta Sarkar and Dr. Sachin Lodha from Tata Consultancy Services for their insightful feedback. We also thank anonymous reviewers for their valuable comments in improvising the paper.

References

1. Satoshi Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2009.
2. King, Sunny, and Scott Nadal. "Ppcoin: Peer-to-peer crypto-currency with proof-of-stake." self-published paper, August 19 (2012).
3. Dziembowski, Stefan, et al. "Proofs of space." Annual Cryptology Conference. Springer, Berlin, Heidelberg, 2015.
4. K Wüst, A Gervais, "Do you need a Blockchain?", IACR Cryptology ePrint Archive 2017, 375.
5. A. Back., "Hashcash: A denial-of-service countermeasure," 2002.
6. Ethereum, `https://www.ethereum.org/`.
7. Castro, M. and Liskov, B., "Practical Byzantine Fault Tolerance", Third Symposium on Operating Systems Design and Implementation (OSDI. USENIX),1999.
8. C. Cachin, S. Schubert, M. Vukolic, "Non-determinism in Byzantine Fault-Tolerant Replication", International Conference on Principles of Distributed Systems (OPODIS), 2016.
9. Identity Mixer, `https://www.zurich.ibm.com/identity_mixer/`.

10. Blockchain in Air ticketing, https://medium.com/@PasschainBlog/is-blockchain-necessary-in-airline-ticketing-d4e089910bd3
11. Hossein Shafagh and Lukas Burkhalter and Anwar Hithnawi and Simon Duquennoy, "Towards Blockchain-based Auditable Storage and Sharing of IoT Data," ACM CCSW 2017.
12. Ahmed Kosba, Andrew Miller, Elaine Shi, Zikai Wen, Charalampos Papamanthou, "Hawk: The Blockchain Model of Cryptography and Privacy-Preserving Smart Contracts," IEEE Symposium on Security and Privacy (SP), 2016.
13. Conner Fromknecht, Dragos Velicanu, Sophia Yakoubov, "A Decentralized Public Key Infrastructure with Identity Retention," https://eprint.iacr.org/2014/803.pdf .
14. Fan Zhang, Ethan Cecchetti, Kyle Croman, Ari Juels and Elaine Shi, "Town Crier: An Authenticated Data Feed for Smart Contracts," ACM SIGSAC Conference on Computer and Communications Security(CCS), 2016.
15. Oraclize, http://www.oraclize.it/#projects.
16. Everledger, https://www.everledger.io/.
17. Certificate Transparency, https://www.certificate-transparency.org/.
18. G. Ateniese, B. Magri, D. Venturi, E. Andrade, "Redactable Blockchain -or- Rewriting History in Bitcoin and Friends," IEEE European Symposium on Security and Privacy (Euro SP), 2017.
19. Elli Androulaki, Artem Barger, Vita Bortnikov, Christian Cachin, Konstantinos Christidis, Angelo De Caro, David Enyeart, Christopher Ferris, Gennady Laventman, Yacov Manevich, Srinivasan Muralidharan, Chet Murthy, Binh Nguyen, Manish Sethi, Gari Singh, Keith Smith, Alessandro Sorniotti, Chrysoula Stathakopoulou, Marko Vukolic, Sharon Weed Cocco, Jason Yellick, "Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains," EuroSys 2018.
20. Supplychain Traceability: Anti-counterfeiting, https://wiki.hyperledger.org/_media/groups/requirements/hyperledger_-_supply_chain_traceability-_anti_counterfeiting.pdf.

View publication stats