Survey of Consensus Protocols on Blockchain Applications

Lakshmi Siva Sankar
TIFAC-CORE in Cyber Security
Amrita School of Engineering, Coimbatore
Amrita Vishwa Vidyapeetham
Amrita University, India
cpslakshmi94@gmail.com

Sindhu M.
TIFAC-CORE in Cyber Security
Amrita School of Engineering, Coimbatore
Amrita Vishwa Vidyapeetham
Amrita University, India
m_sindhu@cb.amrita.edu

M. Sethumadhavan
TIFAC-CORE in Cyber Security
Amrita School of Engineering, Coimbatore
Amrita Vishwa Vidyapeetham
Amrita University, India
m_sethu@cb.amrita.edu

Abstract— **Blockchain is a distributed, transparent, immutable ledger. Consensus protocol forms the core of blockchain. They decide how a blockchain works. With the advent of new possibilities in blockchain technology, researchers are keen to find a well-optimized Byzantine fault tolerant consensus protocol. Creating a global consensus protocol or tailoring a cross-platform plug and play software application for implementation of various consensus protocols are ideas of huge interest. Stellar Consensus Protocol (SCP) is considered to be a global consensus protocol and promises to be Byzantine Fault Tolerant (BFT) by bringing with it the concept of quorum slices and federated byzantine fault tolerance. This consensus's working and its comparison with other protocols that were earlier proposed are analyzed here. Also, hyperledger an open-source project by Linux Foundation which includes implementing the concept of practical byzantine fault tolerance and also a platform where various other consensus protocols and blockchain applications can be deployed in a plug and play manner is also being discussed here. This paper focuses on analyzing these consensus protocols already proposed and their feasibility and efficiency in meeting the characteristics they propose to provide.**

Keywords—Blockchain; Byzantine Fault Tolerance; Consensus Protocol; Quorum Slice

I. INTRODUCTION

Trade has been a part of our day-to-day activities even before the time of industrialization. The famous Barter system, which was to sell or exchange commodities according to ones need. But trade is not just that, it is also based on trust among those who take part in it. Barter system proved to be impractical, as people did not know whom to trust. Thus evolved the system of Triple- entry book keeping [1], where a trusted third party, equally trusted upon by either of those who are involved in the trade will keep record of the transaction details along with the participants of trade. This could provide authenticity, non-repudiation and integrity. But too much trust on third-party did not seem to be a secured way of trade. This is solved cryptographically with the introduction of digital signatures. Authenticity and integrity are provided using elliptic curve digital signature algorithm, thus providing security. But that could not prevent the system from double spending attacks. Double spending occurs when, say a party A exchanges one of his "share Y", with a party B in exchange for a "share X". A being a malicious user cunningly exchanges the same "share Y" with a party C for a "share Z". Since party C is unaware of the transaction between A and B it agrees for this transaction with A. Now A has shares X and Z. This occurred because C did not know about the transaction between A and B.

The solution to this problem was to provide replicas of the transaction and distribute it to all, so that everybody gets to know who all have traded what all. This is the concept of distributed systems, maintains replicas of a data distributed over a network, thus providing integrity. This required the entities of the system to be available all the time a transaction is made. Hurdles like geographical location of various systems, varied time zones etc. also made the proposed idea next to impractical. This brought in the idea of peer-to-peer networks into consideration. Now all the peers could get information on all transactions and also validate them. And the number of peers required to validate could be decided as zero, where the traders validate transaction (they trust each other) or *n*-transaction where minimum of *n* peers must validate the transaction to be confirmed.

But there is a possibility for an attacker to create *n* number of false peers and validate his in valid transaction. This is called the Sybil attack. This lead to borrowing the idea of hashcash, used to filter out spam e-mails, to be brought into blockchains used as "proof-of-work" in case of bitcoin. This involved solving a cryptographically complex problem so as to validate a transaction. Even though the problem is hard to be solved, it must be easy to be verified. The concept of hashcash also required the same property. The simple idea was that only a valid entity would put in effort and use high power for validating a transaction. For a bitcoin transaction, that implements the concept of proof-of-work, for validating a block of transactions it takes minimum of 10 minutes with high-speed processors working simultaneously. Thus it was considered hard to compute the same *n* times to validate a block of transaction. Though this solved the Sybil attack [13] to certain extend, there aroused the problem of byzantine generals problem. That is the assumption that 1/3rd of the peers may be lying or are

978-1-5090-4559-4/17/$31.00©2017IEEE

malicious. One of the best-approved solutions to this problem in distributed networks was the Paxos algorithm [13]. Few of such consensus algorithms are discussed in the following sections.

II.TYPES OF BLOCKCHAIN

The three broad classifications of blockchains are public blockchain, consortium blockchain and fully private blockchain [6].

In case of public blockchain, there is no centralized authority or no party has more power than the rest. Here everyone is open to join or leave as they wish. The blockchain is publicly open and everyone has the right to validate a transaction. Bitcoin is the best example for public blockchain. In case of bitcoins it is the miners, who validate the transaction. They get bitcoins in form of transaction fees and the new bitcoins generated for the effort they put in order to solve the proof of work challenge.

In the case of consortium blockchain, not everyone have equal rights of validation of transactions. Only few people are given certain privileges over validating the transactions. The rest of them may validate, but these selected number of people before the implementation must reach consensus. A slightly different version of this sort of blockchains is the fully private blockchain. It has a centralized structure. A single entity has the power to take decisions and the validation process is also controlled by this entity. The centralized head will make sure that the consensus that is followed is the one it proposed. This is more like having a centralized body like government in different nations. The public blockchain system is also known as permissionless blockchain while the other two classifications come under the category of permissioned blockchain. Permissioned blockchains are faster, more energy efficient and easily implementable compared to permissionless blockchains. Depending upon need and implementation environment, one should decide on which algorithm to deploy. Algorithms based on permissioned blockchain are the most used and more applications based on it are being researched on.

Requirement for a very secure and combat way of storing data have made way to the well awaited project Storj [10], a cloud based blockchain implementation that stores tons of data securely. Also, Corda [8] is another blockchain implementation that is used to automate financial documents. There are also other blockchain applications like Tezo [7] that claims to be self-amending blockchain application. The consensus protocol in this application is not just for validation but also for upgrading the system itself. Music artists can record their work into a decentralized music platform Ujo. This is a blockchain and it is recorded as smart contracts [6].

III.CONSENSUS ALGORITHM

Literally consensus mean is agreement. Consensus algorithms are those algorithms that help a distributed or decentralized network to unanimously take a decision whenever necessary. Its features include assuring decentralized governance, quorum structure, authentication, integrity, non- repudiation, byzantine fault tolerance and performance [11]. The public blockchain, bitcoin uses the concept of "proof of work". There are other forms of consensus protocols applying the concept of Proof of Stake (PoS), Proof of Elapsed Time (PoET), Proof of Existence (PoE), Delegated Proof of Stake (DPoS), Proof of Activity (hybrid of proof of work and proof of stake), Proof of Importance, proof of storage [6] etc.

In the case of bitcoins, a consensus algorithm help deciding the validity of the transactions and also helps avoiding the forking problem in blockchain. Forking problem occurs when two miners simultaneously mines a block of transaction. Now a fork occurs in the otherwise linear form of blockchain. This situation is prevailed by the concept of longest chain rule. Thus the consensus protocol manages to avoid malfunctioning of the blockchain architecture. Similar consensus algorithms that pertain the blockchain architecture and ensure its proper functioning like Stellar Consensus Protocol, Corda and Hyperledger are studied and analyzed in the following sections.

Stellar Consensus Protocol

Stellar Consensus Protocol was proposed to create an open-source platform that allows users to create applications using blockchain architecture. Stellar has taken on the challenge of providing integrated microfinance services all across Nigeria. Since February 2016, Stellar has been running a nation-wide test network in co-operation with the micro financing software provider Oradian. About 90 % of the current customers are female. This will contribute to significant economic empowerment of women in the developing world [6, 15].

Stellar Consensus Protocol (SCP) [3] proposed by David Mazieres, follows federated byzantine fault tolerance (FBFT). They introduced the concept of quorum slices. A quorum is a set of nodes that act together to attain consensus and a quorum slice is its subset, which helps a node in its process of agreement. SCP is a global consensus protocol, which consists of nomination protocol and ballot protocol.

Initially the nomination protocol is run. During this, new values called candidate values are proposed for agreement. Each node receiving these values will vote for a single value among these. Eventually it results in unanimously selected values for that slot.

After successful execution of nomination protocol, the nodes deploy the ballot protocol. This involves the federated voting to either commit or abort the values resulted from nomination protocol. This results in externalizing the ballot for the current slot. The aborted ballots are now declared irrelevant. But there can be stuck states where nodes cannot reach a conclusion, whether to abort or commit a value. This situation is avoided by moving it to a higher valued ballot, considering it in a new ballot protocol execution. This helps in case a node believes that this stuck ballot was committed. Thus SCP assures avoidance and management of stuck states and thus provides liveliness.

SCP protocol claims to be free of blocked states, provides decentralized control, asymptotic security, flexible trust and low latency. But it does not guarantee safety all the time. If the user node chooses an inefficient quorum slice security is not guaranteed.

The key distinction between FBA and prior Byzantine agreement systems is that, in case of FBA, the nodes that involve in the transaction decide quorums. SCP in spite of a FBA construction achieves optimal safety against ill-behaved participants [3].

Corda

It is a blockchain application that focuses on controlling financial contracts as well as maintains records of them. They include legal documents modeled as proposed by law, which are based on computer codes and are automatically generated. These legal documents contain the information of rights of the individual mentioned. This technology could revolutionize the financial and legal sectors in many countries.

These documents that are automatically generated are termed as state objects [8]. So, a state object encompasses the details on who are involved in the transaction, timestamp information, how long it is valid, the alterations made and the current state of the transaction. They are confidential and can be viewed only by those who have rights to do so. Thus the openness feature of blockchains is not adopted in case of corda. The legal documents of transaction are visible only to those who where involved in the transaction and the hash values are used to ensure this. The main characteristics of corda are, automated smart contracts, time stamping the documents to ensure uniqueness and a self-sufficient framework of prebuilt document templates. They provide a plug and play environment for its users. This helps the user to implement their own smart contract code thus allowing them to use an efficient algorithm according to their need and this helps them to maintain privacy. The new legally stable smart contracts thus proposed and implemented by the users will allow the system to be more scalable.

Consensus of the corda involves acquiring the values currently available, combine it with smart contracts and produce new results or states. The two key aspects that are followed to attain consensus are t*ransaction validity* and *transaction uniqueness* [8]. Checking the validity of the smart contract code used and whether it was run with appropriate signatures, resulting in error free execution, maintain the transaction's validity. The time stamping and other constraints involved in execution of smart contracts allows maintaining uniqueness of the transaction.

Hyperledger Fabric

Hyperledger [2] was started as a project under the Linux Foundation in early 2016. It aimed at creating an open-source cross-industry standard platform for distributed ledgers. Hyperledger Fabric is an implementation of a distributed ledger platform for running smart contracts, leveraging familiar and proven technologies, with a modular architecture al- lowing pluggable implementations of various functions. The distributed ledger protocol of the fabric is run on the peers [2]. The fabric distinguishes peers as:

***Validating peer*:** These are peers that run the consensus algorithm, thus validating the transactions.

Non-validating peer: These are nodes that act as a proxy that helps in connecting clients to validating peers. They are not capable of executing any transaction but can verify them.

The validating peers run a BFT consensus protocol for executing a replicated state machine that accepts three types of transactions as operations:

Deploy transaction: The transaction involves accepting the code of the smart contract to be deployed (chaincode) as parameter. The chaincode is written in Go language and is installed on the peers.

Invoke transaction This involves accepting arguments specific to the activity that is performed by the transaction and executing the chaincode. The result of execution, whether successful or not is indicated.

Query transaction: It returns an entry of the state directly from reading the peers persistent state and may not ensure linearizability.

In case of hyperledger fabric, the blockchains hash chain is computed based on the executed transactions and resulting persistent state. The replicated execution of chaincode is used for validating the transactions. They assume that among n validating peers, at most $f<n/3$ (where f is the number of faulty nodes and n is the number of nodes present in the network) may behave arbitrarily, while others will execute correctly, thus adapting to concept BFT consensus. Since hyperledger fabric proposes to follow Practical Byzantine Fault Tolerance (PBFT), the chaincode transactions must be deterministic in nature, otherwise different peers might have different persistent state. SIEVE protocol is used to filter out the non-deterministic transactions, thus assuring a unique

persistent state among peers.

It supports enrollment and transaction authorization through public key certificates, and confidentiality for chaincode realized through in-band encryption. More precisely, for connecting to the network every peer needs to obtain an enrollment certificate from an enrollment Certification Authority (CA) that is part of the membership services. It authorizes a peer to connect to the network and to acquire transaction certificates, which are needed to submit transactions. Transaction certificates are issued by a transaction CA and support pseudonymous authorization for the peers submitting transactions, in the sense that multiple transaction certificates issued to the same peer (that is, to the same enrollment certificate) cannot be linked with each other. Confidentiality for chaincodes and state is provided through symmetric key encryption of transactions and states with a blockchain specific key that is available to all peers with an enrollment certificate for the blockchain. Extending the encryption mechanisms towards more fine-grained confidentiality for transactions and state entries is planned for a future version.

Membership among the validating nodes running BFT consensus is currently static and the setup requires manual intervention. Support for dynamically changing the set of nodes running consensus is also included in their work to do. Sawtooth Lake [12], an initiative by Intel, a blockchain application, is announced to be exercised using hyperledger.

IV. ANALYSIS OF STELLAR CONSENSUS PROTOCOL AND HYPERLEDGER

The idea of hyperledger-fabric simplifies the process of building a blockchain application and deploying it. The hyperledger-fabric platform initially proposes to use the PBFT consensus protocol. This requires the transaction to be in a deterministic state to be confirmed. This assures security but the FBFT approach adopted by SCP provides more flexibility of usage for the users. In case of corda, the users can add in their own smart contract code as in Hyperledger, but all cannot view their transactions. Though blockchains are claimed to be open distributed ledgers, Corda [8] does not provide openness. This helps it to be more secure as only authorized access and view ability of contracts are allowed. Hyperledger uses Go language for its implementation [14]. But the platform supports the use of blockchain applications written in other programming languages by adding it to its chaincode module. The concept of quorum slices in case of SCP provides asymptotic security and flexible trust, making it more acceptable than other earlier consensus algorithms utilizing FBFT, like the ripple consensus protocol [4]. Here the user is provided more independence in deciding whom to trust. Though the concept of asymptotic security was included in the tendermint consensus algorithm [5] proposed earlier, SCP's flexible trust lets it stand different from the previously proposed consensus protocols [15]. The proof of work algorithms results in huge latency and computational power. SCP and hyperledger assures much less latency in operation. Similarly, since Corda's states are not accessible to all, they are able to maintain low latency rates. Lately, hyperledger proposes to avoid use of REST APIs so as to increase speed of transaction validation. Hyperledger being a platform for blockchain applications facilitates various consensus protocols to be deployed. The Sawtooth Lake project, which proposed to be deployed on hyperledger, uses the PoET consensus protocol for validation and uses python for its implementation. Thus various consensus protocols have different characteristics, which are to be looked upon by the users considering their requirements.

V. CONCLUSION

Blockchain technology is expected to revolutionize the finance and banking sectors around the world. Many banks have already started building their own blockchain application or are finding ways to initiate one. In this paper, a comparative analysis of consensus protocol on SCP, corda and hyperledger are made. The SCP has already gained popularity for its ability to connect people of various trends in the society. SCP proposes to provide asymptotic security and flexible trust by introducing the concept of quorum slices that ensures more freedom to the users on deciding the participants they need to trust for validating their transactions. It also manages the stuck situation that could occur during consensus by neutralizing them. But, as mentioned in SCP [3], when providing the users more freedom in choosing the nodes to trust, the system may not be able to provide the user with all its features if the user is new to the system and does not choose the nodes to trust efficiently. Corda maintains records of various business and financial contracts. These may not be open to all, but in financial institutions where data is confidential, this applicability ensures privacy. Since, the contracts are automated and they follow the then legal format for all its documents, they act as a real time saver. Also its scalability is huge, allowing institutions from various sectors to implement and maintain their financial records using Corda. Hyperledger project allows various blockchain technologies to interconnect and assures a secure plug and play environment for them. The hyperledger does not provide the users as much freedom as the SCP does. The current programming languages supported as chaincode in hyperledger include mainly java, Go. Hyperledger thus focus more on interoperability of various proposed blockchain applications, but support for various other programming languages like Haskell, Perl etc. are proposed but yet to be implemented. The applicability of blockchain technology is not confined to bank or finance sector. New innovations have already come out of the blockchain technology. Blockchain technology is now used for various IoT applications too to secure them against intruders. Consensus protocol being the working entity of blockchain can thus be of varied implementation styles. An efficient protocol could thus produce tremendous results to the growth of economy.

REFERENCES

[1]. Andreas M.Antonopoulos, Mastering bitcoins, 2014.

[2]. Christian Cachin, Architecture of the Hyperledger Blockchain Fabric, July 2016.

[3]. David Mazieres,, The Stellar Consensus Protocol:A Federated Model for Internet-level Consensus , February 25,2016.

[4]. David Schwartz, Noah Youngs, Arthur Britto, The Ripple Protocol Consensus Algorithm, 2014.

[5]. Jae Kwon, Tendermint: Consensus without Mining, 2014.

[6]. Juri Mattila, The Blockchain Phenomenon, Berkeley Roundtable on the International Economy (BRIE) University of California, September 2, 2014.

[7]. L.M Goodman, Tezos a self-amending crypto-ledger,September 2, 2014.

[8]. RichardGendalBrown, JamesCarlyle, IanGrigg, MikeHearn, Corda: An Introduction, August, 2016.

[9]. Satoshi Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System, 2008.

[10]. ShawnWilkinson, TomeBoshevski, JoshBrandoff and VitalikButerin, Storj A Peer-to-Peer Cloud Storage Network,December 15, 2014.

[11]. Sigrid Seibold, George Samman, Consensus Immutable agreement for the Internet of value, 2016.

[12]. https://intelledger.github.io/introduction.html

[13]. https : //www.igvita.com/

[14]. http: //www.research.ibm.com/labs/zurich/

[15]. https: //www.stellar.org/