Blockchain Architecture and Design
Andrew Yiyun Zhu

Introduction

The research area of this paper is on blockchain architecture and design. Given the rise in the cryptocurrency market in the past few years, there exists significant public interest in the technology behind blockchain design which is at the core of cryptocurrency. The research dives into what is blockchain, what is the architecture and design, current applicability, and potential ways it could shift existing software infrastructures. The analysis is based on three papers, "Blockchains and smart contracts for the IoT" provides an overview on blockchains and then indulges into the applicability for the IoT. "The Blockchain as a Software Connector" is on the design of blockchain architecture and its applicability through sample experiments. Lastly, "A Blockchain-Based Architecture for Collaborative DDoS Mitigation with Smart Contracts" is on how a blockchain based design could be used to improve existing DDoS mitigation systems.

Problem Statement

Existing architectures for distributed peer-to-peer networks consists of untrusting members who interact with each other under an intermediary [1]. Examples span across multiple sectors such as in Banking (banks are intermediary), real estate (agents are intermediaries), healthcare (hospitals), et al. The problem is many of these agents cannot always be trusted. Instances of fraud and lack of transparency among intermediaries is prevalent across multiple sectors. Examples include 2008 monetary crisis of banks over-lending, or the 2017 Equifax breach of consumer information [2]. Blockchain architecture was created with the idea of eliminating the central authority by building a decentralized network. Applications that previously run only through a trusted intermediary can now operate in a decentralized fashion without a central authority.

Summary of Blockchains and Smart Contracts for the Internet of Things

The research paper summarizes blockchain and smart contract design while revealing how it could be applicable towards the IoT, and cause significant transformations across several industries. A blockchain is a immutable data structure that is replicated and shared among a decentralized distributed network. It's a log of records batched into timestamped blocks. Each block is identified by a cryptographic hash and each block references the block that came before it. The blockchain network runs through a series of nodes (be a User or Computer) that form a peer-to-peer network. Users can interact with the network by initiating transactions by signing it with their private key. Upon signing, users then broadcast their public key to the network. Neighbor users then validate the transaction and it moves towards the "mining" node if the transaction is valid based on a set of predetermined rules [1]. A simple example in the case of Bitcoin would be nodes validate that the transaction as valid if there are sufficient funds. Nodes verify transition of Person1(send 1 bitcoin to person 2) that person 1 needs to have a minimum of 1 bitcoin in their state before the new state can be applied, e.g. Transaction is deemed valid and Person 1 new state = old state – 1 bitcoin, Person 2 new state = old state + 1 bitcoin. The mining node then collects multiple transactions that they package into a timestamped block. The block consists of their own hash header, the hash to the previous block, a timestamp, and the list of transactions. Once the miner collects all information, it then broadcasts to the network the block result. If the network nodes reach consensus by validating the block as correct, the block gets

added to the distributed ledger known as the blockchain and the process repeats for the next blocks to be added to the network blockchain [1]. The issue at hand is that any node can assemble the next block as long as they solve the hash code that references the previous block. A malicious attacker could assemble their own block, convince or team-up with other attackers to validate their block and create a false blockchain by double spending. Bitcoin gets around this issue by a concept known as Proof of Work. Essentially for the block to be broadcasted, miners need to solve the SHA-256 header which is computationally expensive. This makes it impossible for malicious users since they would need to have more computational power than the entire network to falsify the blockchain [1].

Smart contracts are based on top of blockchain technology and exists as scripts that are like stored procedures on relational databases. They can be deployed inside of the blockchain environment as scripts that only get executed when a specific condition is met [1]. These contracts allow business logic and multi-step processes to occur between counterparties without the need for an intermediary. For example, Person 1 wants to trade 5 units of ‘A’ for 5 units of ‘B’, this script is then placed into the virtual environment. If another person comes along and wants to trade 5 units of B for 5 units of A. The transaction automatically gets triggered and is verified by nodes and added to the blockchain.

Current applicability of blockchain is already widely used in the world of digital currency. Ethereum is an adopter of smart contracts while Bitcoin is the pioneer behind the blockchain. technology. In the case of the IoT, a blockchain design could alleviate high maintenance costs of software updates and upgrades of the millions of devices in use. Additionally, transparency is an issue amongst consumers. Blockchain solves transparency problems as a public ledger where anyone can access all the transactions in the block. IoT devices could have smart contracts that seek out software updates rather than device calls to the server that results in continuous 404 errors when updates are not found [1]. In supply chain and logistics, we could move assets between checkpoints and ports through a series of smart contracts and validations without the need for human intervention at each port. Nevertheless, the design of blockchains and smart contracts come with limitations. Compared to a centralized database, blockchains perform with much lower throughput and higher latencies. As a result, scalability is a significant concern given how only a limited amount of transactions can be included in each block [1]. Privacy is also a complicated issue. Even though transactions are referenced by their public key, users could begin to see patterns of certain addresses given the transparency of the system. Miners need to be carefully selected because even though they cannot rewrite a one-way blockchain design, they can still reject a valid transaction. Smart contracts cannot be altered or undone, therefore it’s important that they ‘self-destruct()’ upon completion or they could have significant dead end provisions inside the virtual environment. Overall, the combination of blockchains and smart contracts is anticipated to cause significant transformations across several industries in the IoT [1].

Critique and Reflection on Blockchains and Smart Contracts for the Internet of Things

I believe the paper does a respectable job of detailing the description of how blockchains and smart contracts work from a high-level and application level. They followed their guidelines and their statement of evaluating blockchains and smart contracts in addition to applicability for the IoT. However, they are notably biased towards the positivity of blockchain technology and do

not reflect in the short-comings on solving scalability issues before blockchain can be adopted on a wide-scale basis. I understand that they state their deployment issues briefly on privacy and scalability, but no measures or suggestions are noted to improve scalability for the long-term. In the conclusion, they mention the power of blockchain technology, which I believe is acceptable given the transforming nature and potential it could have on several industries. Nevertheless, they express their opinion on significant transformations for blockchain and IoT without suggesting a single viable solution or direction to a fundamental component for adoption, which is scalability. As of current times, Bitcoin and Ethereum can handle on average 3-20 transactions per second, while VISA can handle on average 2000 transactions per second [2]. I believe the future direction of scalability needs to be addressed before concluding that blockchains will have significant transformations for IOT.

Summary The Blockchain as a Software Connector

This paper evaluates how Blockchain serves as a software connector by evaluating the performance of quality attributes. Based on personal projects that the authors are undertaking, rationales are then provided as to whether to support architectural design of decentralized blockchains compared to traditional centralized data storage.

Software connectors serves as a fundamental building block and interaction mechanism for components. Blockchains provide software connector capabilities through communication and coordination. Blockchains can be further abstracted into two layers, blockchain layer which hosts implementation and management such as smart contracts. The second layer would be application layer which hosts off-chain data and off-chain control such as off-line data and application logic. Blockchains are able to communicate between components in the architecture seamlessly. Bitcoin adds data to transactions while Ethereum adds data into contract storage. Transactions once included in the blockchain are communicated across the network to all components. In addition, different components that aren't connected can communicate and coordinate through the blockchain by submitting multi-step processes with smart contracts [2]. There exist limitations with blockchain as a software connector, notably scalability. Improvements consider increasing the block-size to include more transactions per block, or eliminating key signatures, or using third party as verification to alleviate the load on miners. Since the execution environment of the blockchain is a closed environment which cannot import external states through polling external servers, a validation oracle is a suggestion to include human verification. Blockchain as a connector could be used both in a public or private network. Public network, anyone can act as miners and have permission-less capabilities. On the other hand, in a private network, only some are assigned certain capabilities and thus it's a permissioned network [2].

Blockchains are different than centralized traditional data stores since they do not rewrite or append but only create new transactions. Blockchains are more sustainable data storages since data is duplicated on every node, but the throughput and latency underperforms. [2]. The team undertakes two projects by using blockchain architecture. The first exists as data monetization where owners sell data to consumers. Dataset and payment along with logs are held in the blockchain, while analytics and raw data is hosted off chain in addition to policy enforcement. The second project exists as a platform to support organizations sharing sensitive data. The key holding the encrypted value of the contract is held in the blockchain along with access control, while the documents and identities of all participants are held off chain in addition to all

negotiation and authentication documents. From these projects, the team discovered scalability and performance is very limited for blockchains [2]. Only private blockchains can guarantee data privacy. The miners are always given an incentive through fees to continue validating the transactions with computational power. It is concluded the design of blockchain provides communication and coordination services to all components in software architecture landscape. Design considerations will vary depending on the type of distributed environment.

Critique and Reflection on The Blockchain as a Software Connector

This paper is clear and well presented on their problem statement which is presenting the architectural design decisions on blockchain architecture for a specific software solution. They addressed various scenarios, tradeoffs, and design considerations for using blockchain as a software connector based on the type of project involved. Notably, permissionless vs permissioned blockchain given the security and sensitivity of data involved. Despite clear and thoughtful research, I disagree with the methodology in which they use Blockchain as a software connector. They utilize 2 layers, application layer and blockchain layer. If you include the application layer which they use in their both their projects, in addition to the validation oracle, it defeats the purpose of decentralization. The blockchain's core design is to facilitate transactions without a 'trusted owner', or centralized environment. If you include off chain services, or even going so far to say, "the government is a trusted party anyway, thus, we use government as validation oracle that injects external state into the blockchain." [2 pg. 9/10]. It eliminates fundamentals that blockchain was founded on. Using off chain storage and services may improve privacy and functionality by using blockchain design to connect components from within the blockchain layer and outside of it, but then we are steering away from blockchain's decentralized design and going towards a traditional approach of architectural design for data storage. This paper presents projects that use blockchain as a software connector in existing centralized systems rather than using blockchain in a decentralized system as a software connector.

Summary A Blockchain-Based Architecture for Collaborative DDoS Mitigation with Smart Contracts

The problem is a rapid growth in the number of DDoS (Distributed Denial-of-Service) attacks based on the increase in traffic volume across networks. Existing defense mechanisms to mitigate DDoS are proposed but only a few are considered for adoption based on either effectiveness or implementation complexities [3]. IETF (Internet Engineering Task Force) proposes DOTS (DDoS Open Threat Signaling) which aims to broadcast attackers IP to both intra-AS and inter-AS networks. However, the downfall is communication across ISP's will be difficult because some don't have business relationships with each other and are competitors. Third parties are also suggested as intermediaries for traffic redirection to filtering out attacker's IP addresses but those come with significant implementation difficulties. The proposed architecture from the paper involves participants in a network to create smart contracts where the blockchain acts as a list of prohibited addresses of attackers [3]. The design has three main components, 1) customers report IP to blockchain via smart contracts. 2) ASes (Autonomous Systems) publish and broadcasts the blacklisted IP addresses and may implement their DDoS mitigation mechanisms. 3) Blockchain and Smart Contracts via the Ethereum virtual machine environment runs solidity smart contracts which has the logic to report IP addresses into transactions within the blockchain.

The proposed solution allows multiple domains from various ISPs to successfully communicate by using smart contracts as flags for attacker IPs. The solution can be widely deployed on top of existing DDoS mitigation techniques without starting from scratch by adding in the feature of smart contracts in the Ethereum virtual machine. A downfall is that the solution works for small number of attacks, but costs could greatly increase for substantial number of attacks based on the fee of each smart contract transaction to be paid. Considerations for future work include ways to allocate responsibility if one AS is targeted more than others in addition to accounting for dynamic IP attackers [3].

Critique and Reflection A Blockchain-Based Architecture for Collaborative DDoS Mitigation with Smart Contracts

The paper does a fantastic job of clearly stating their problem statement, their solution, and concisely describing each in efficient but accurate detail. Of the three papers, I feel they are the most transparent and unbiased towards their stance based on their content and their explanations. Although they are supportive of smart-contracts and how it can be deployed to existing DDoS architectures, they state the considerations for the future and how it can be scaled unlike the previous two papers. For example, even though a limitation is blocking only for static IPs, future considerations could check for dynamic IPs first. In addition, their future considerations for an AS being targeted more than others is more so a business problem rather than a design or architecture problem as was the case for blockchain scalability in the first two papers.

Conclusion

Blockchain design and architecture has the potential to change many industries. Decentralization would be favored by more participants than rejected based on the amount of money saved as that is what seems to drive everything. In addition, efficiency drives the technological advancements and that is one area where blockchain is significantly lacking in scalability. Scalability and privacy will be the most concerning aspects moving forward before wide-stream adaptation. Given the amount of public interest in blockchain technology based on bitcoin and cryptocurrency in general, I am confident scalability will eventually be addressed. However, based on current events and results from these research studies, seldom individuals propose any long-term solutions. Only time will tell if scalability is solved, and if so, we could potentially see a world that is run on blockchain without banks and without intermediaries.

Works Cited List (APA)

1. Christidis, K., & Devetsikiotis, M. (2016). Blockchains and smart contracts for the internet of things. *IEEE Access*, *4*, 2292-2303.

2. Xu, X., Pautasso, C., Zhu, L., Gramoli, V., Ponomarev, A., Tran, A. B., & Chen, S. (2016, April). The blockchain as a software connector. In *Software Architecture (WICSA), 2016 13th Working IEEE/IFIP Conference on* (pp. 182-191). IEEE.

3. Rodrigues, B., Bocek, T., Lareida, A., Hausheer, D., Rafati, S., & Stiller, B. (2017, July). A Blockchain-Based Architecture for Collaborative DDoS Mitigation with Smart Contracts. In *IFIP International Conference on Autonomous Infrastructure, Management and Security* (pp. 16-29).