# A Scalable Blockchain Framework for Secure Transactions in IoT

Sujit Biswas, Kashif Shaif, *Member, IEEE,* Fan Li, *Member, IEEE,* Boubakr Nour, and Yu Wang, *Fellow, IEEE*

***Abstract*—Internet of Things and Blockchain technologies have been dominating their respective research domains for some time. IoT offers automation at the finest level in different fields, while Blockchain provides secure transaction processing for asset exchanges. The capability of IoT devices to generate transactions prompts their integration with Blockchain as the next logical step. The biggest challenges in this integration are the scalability of ledger and rate of transaction execution in Blockchain. On one hand, due to their large numbers, IoT devices will generate transactions at a rate which current block chain solutions cannot handle. On the other hand, implementing Blockchain peers onto IoT devices is impossible due to resource constraints. This prohibits direct integration of both technologies in their current state. In this article, we propose a solution to address these challenges by using a local peer network to bridge the gap. It restricts the number of transactions which enters the global Blockchain by implementing a scalable local ledger, without compromising on the peer validation of transactions at local and global level. The testbed evaluations show significant reduction in the block weight and ledger size on global peers. The solution also indirectly improves the transaction processing rate of all peers due to load distribution.**

***Index Terms*—Internet of Things, Blockchain, Scalability, Security, Transaction rate, Ledger size**

## I. Introduction

INTERNET of Things (IoT) provides a platform for connecting daily use smart devices to gather, share, and forward information. Many of these exchanges are financial transactions which will dominate the future Internet architecture. The upcoming 5G technology will provide special support for machine-to-machine (M2M) [1] communications, which will allow unprecedented growth for IoT. Some estimates approximate that 50 billion devices will be registered as IoT devices by 2020 [2], and will generate trillions of transactions every day. Currently most of communication is performed based on a client-server model. In a centralize communication model, the system administrator may disclose sensitive data (e.g. health care, finance, etc.) due to insider attacks [3]. Furthermore, the conventionally centralized computing model favors several large-sized distributed data centers which creates a huge burden for computing, storage, and networking resources [4]. However, using traditional centralized communication models for such large scale data communication, storage, and analysis, from billions of devices is next to impossible. Although cloud and edge/fog architectures [5] do provide virtually unlimited storage and processing capacity, the bandwidth required to upload transactions creates a bottleneck in the network.

IoT networks and the embedded devices in them have drastically increased, which presents new dimensions to various threats related to security and privacy. Considering the limitations of existing client-server and cloud technologies, combined with rapid scalability of IoT, many researchers have suggested using Blockchain (BC) as a potential solution for security and privacy issues. The prime motivation for this stems from Bitcoin [6] (a public Blockchain for cryptocurrency transaction), to address the challenges of IoT security.

### A. Blockchain Fundamentals

Blockchain was first introduced as Bitcoin [6] in 2009. As public Blockchain, Bitcoin is the first trust-less peer-to-peer electronic cash which has approximately 28.5 million [7] electronic wallets. Many others electric cash (e.g. ether [8], XRP [9], etc.) have been introduced since, and the number of wallets have increased multi-fold. Blockchain allows value exchange (i.e. transactions) without the need of trust authority from a central entity. These transactions are stored in a ledger which is maintained by a group of connected computers (i.e. peers), unlike a centralized entity like a bank database. BC system performs an autonomous verification (i.e. endorsement) before approving the transaction which plays a key role in ensuring security. The specialty of BC is that it is designed in such a way that no trust is needed, and security & reliability are obtained via special mathematical functions or code. Till 2016, most of the Blockchain networks have been used for cryptocurrency transactions. Recently Blockchain uses have gone beyond cryptocurrency and are beginning to be utilized in other application domains (e.g. IoT, AI, etc.).

**Crypto Blockchain:** These are mainly intended for crypto currencies, which only allows virtual money transactions as a replacement of physical money. The transaction structure is fixed which only carries information regarding the amount of currency being traded. These are usually public chains to

The work of Fan Li is partially supported by the National Natural Science Foundation of China (NSFC) under Grant No. 61772077, 61370192 and 61432015. The work of Yu Wang is partially supported by the US National Science Foundation under Grant No. CNS-1343355, the National Natural Science Foundation of China under Grant No. 61572347, and by the U.S. Department of Transportation Center for Advanced Multimodal Mobility Solutions and Education.

S. Biswas, K. Sharif, F. Li, and B. Nour are with School of Computer Science, Beijing Institute of Technology, & Beijing Engineering Research Center of High Volume Language Information Processing and Cloud Computing Applications, Beijing, China. (e-mail: {sujitedu,kashif,fli,n.boubakr}@bit.edu.cn)

Y. Wang is with Department of Computer Science, University of North Carolina at Charlotte, NC, USA. (e-mail: yu.wang@uncc.edu)

Drs. Sharif & Li are co-corresponding authors.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2018.2874095, IEEE Internet of Things Journal



improve transparency, but lately private crypto chains have also become available. A miner is used for transaction security, which solves a complex mathematical puzzle to ensure the security of blocks as well as transactions.

**Business Blockchain:** In contrast to previous ones, this newer smart contract based blockchain is being considered for a variety of transaction types ranging from ubiquitous devices, real-time based operation management for industrial productions, and intra-application data transfer including financial trades, etc. These are mostly private or permissioned chains. Moreover, instead of a miner, an orderer is used for ensuring delivery guaranties and block creation.

For adoption of business Blockchain with new application cases, significant modifications are necessary to the generic Blockchain protocol.

In light of this, a new permissioned Blockchain, Hyperledger has been developed for a variety of networks. Hyperledger [10] introduces six business frameworks: Fabric [11], Burrow, Iroha, Sawtooth, Indy, and Quilt. Depending on technological requirements and wide variety of consensus algorithm [12], different frameworks can be utilized. Hyperledger *Fabric* can be used as a foundation for developing Blockchain solutions targeted for IoT network. The basic components of a permissioned Blockchain architecture has been shown in Fig. 1. BC network is formed through interconnection of peers, where peers are independent servers. They are responsible for validation and endorsement of transactions, and maintain the distributed ledger. Validation and endorsement process are mostly dependent on smart contract (chaincode) which must be installed on every endorsing peer. Basically, smart contract is a programmatic code to define the transaction rule or policy in between sender and receiver. Successfully endorsed transactions are stored as a block into a common ledger which is integrated with peers. Many transactions may fit into a single block which must be linked with the last block of the ledger by hash value, which makes a chain of blocks. Quantity of transactions per block and block forming time varies depending on orderer's batch timeout configuration. Ordering is intended to provide an atomic broadcast ordering service for consumption by the peers. Orderer and Membership Service Provider (MSP) provide block creation and user services respectively, to all the peers in a business Blockchain. Table I lists the key principles of BC system.

TABLE I: Basic properties of Blockchain

| Property | Details |
|---|---|
| Decentralized Control | No single authority will control the system or rules. |
| Data Transparency | Transactions can be read by anyone based on smart contract policy. |
| Consensus | Transactions are validated by all endorsing peers in the network. If any of the endorser somehow fails to endorse, transaction will be denied (depending of endorsement policy). |
| Distributed Information | Successful transaction payload will be stored into all peers simultaneously. |
| Secure | It is an immutable distributed database which cannot be controlled by any single entity and has no single point of failure. |

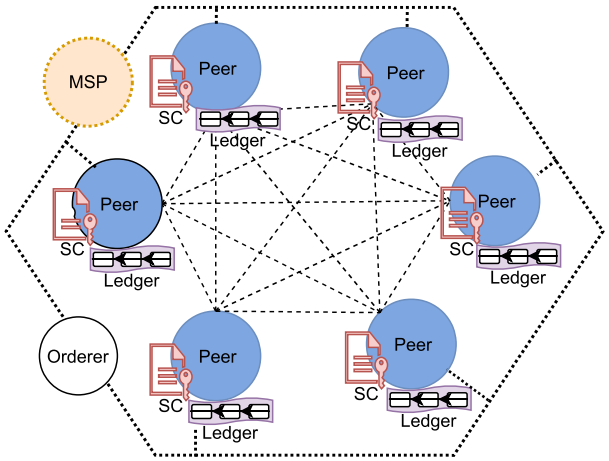



Fig. 1: Permissioned Blockchain: Component interaction.

### B. IoT and Blockchain

It can be expected that the technology behind Blockchain can serve as a basis to keep a ledger of IoT device's transaction logs and communications. IoT structure for Device-to-Device (D2D) [13] systems, has been discussed in literature, hence extending it with Blockchain will provide three key benefits: i.e. Trust (build trust between parties and devices, and reduce the risk of collusion and tampering), reduced costs (remove overhead associated with middlemen and intermediaries), accelerated transaction rate (reduced settlement time).

In order to merge IoT and Blockchain, it is important to understand how BC networks are formed. There can be two models for IoT devices:

1) Each IoT device becomes part of the BC network by implementing a peer client on itself.
2) A number of devices are grouped together (through a well defined mechanism), and use a single peer in global Blockchain to represent them.

The first solution can not be used, as the resources (processing, memory, etc.) required to become a peer, are not available in most IoT devices. However, some devices such as automated machines, robots, etc. may have such capabilities and can become peers, but this would be a subset of all IoT devices. The second solution is more plausible, but current Blockchain networks do not implement such a mechanism. Even if such a mechanism is provided, the scalability of existing BC networks is a major challenge. In order to get an understanding of the scale, assume that 50 billions IoT devices generate on an average 5 transactions per day, which is well below what may happen in reality. The volume of transactions will be approximately 250 billion per day. In a more realistic example, assume a city municipality intends to run a central Blockchain network for different governmental agencies (parking services, toll payment, etc.) in the city. Assume the number of agencies is 50, and each agency has at least 10 peers, where each peer controls 1000 devices or users. Hence, if every user or device generates 10 transactions per day, then from only one agency there will be $10^5$ transactions, and a total of 5 million transactions per day from all agencies.

In light of this, there are two major issues in existing BC solutions; transaction per second (TPS), and ledger storage requirements. Blockchain is fundamentally not designed for high transaction rates, as they will be in IoT. TPS issue has been addressed to some extent with the release of Hyperledger Fabric 1.0, as developers are allowed to customize their endorsement policy and ordering services according to their network requirements. But in any customized situation, generated block will be added to the all peers. If the Blockchain creates single block for every 500 transactions then, it requires 23 GB memory per day or $\approx$ 8.4 TB per year, where wight of a single block with single transaction is at least 4.6 KB (based on our experimental analysis). As the ledger data cannot be deleted or altered, hence this problem is magnified over time. In order to reap the security benefits of Blockchain in IoT, the challenge of efficient ledger size management and transactions per second scalability must be addressed.

In this article, we propose a Blockchain based framework for IoT, for inter and intra organizational transactions where all IoT devices have an association to an organization, through a registration process enabled by a local Certification Authority (CA). In addition, rather than using a peer which is part of the global Blockchain network, we propose a local peer (Lpeer) structure to interact with an associated anchor peer in the global network. This framework aims to solve two main issues, 1) indirect increase in TPS for the global Blockchain network, and 2) limit the geometrical increase of ledger storage requirements at each peer. By using a localized peer for intra-organizational transaction (while maintaining peer validation), the framework limits the ledger size, and distributes it between Lpeer and anchor peer. Inter-organizational transactions are validated through global Blockchain peer network, with 100% peer validation.

## II. Related works

Blockchain technology was designed and has been primarily limited to cryptocurrency solutions (e.g. Bitcoin, Ethereum, etc.). Hyperledger [10] aims to expand BC technology into business networks. Fabric v0.6 for public use had scalability issues which to some extent have been alleviated by Fabric v1.0. It allows the transaction rate to reach $10^4$ TPS, as peers can execute transactions in parallel [14]. In certain application scenarios, this increase may be satisfactory. But with the sheer number of IoT devices, a much higher transaction rate is required. Moreover, the storage requirement of ledger is still an open research issue.

The work in [15], proposes a decentralized access management system for IoT, where access control information is stored and distributed using a Blockchain. By using a gateway called management hub, IoT is connected to BC network. The architecture allows the IoT node to query BC node without endorsement or a well defined device identification mechanism. Certain use cases may require a permission mechanism for such query mechanism. Moreover, the design puts extra burden of transaction fee processing on manager node for millions of transactions per day, while being an external entity to BC network.

A lightweight scalable BC for IoT (LSB) [16] is available online (unpublished) which introduces a centralized manager known as Block Manager (BM), similar to hub, which provides a shared key among home devices and controls all incoming and outgoing transactions requests. Moreover, it stores all the transactions locally. This work does not elaborate how the transactions will be synchronized between local BM and overlay BM. The storage and scalability issues at higher levels are also not addressed. In [11], IBM researchers show 3500 TPS in Hyperledger implementation experiment. They also describe block size variation from 0.5 MB to 4.0 MB (depending on configuration and endorsement policy). Although the paper does not discuss reducing the overhead of block size, storage, and controlling ledger scalability, it does show the need of improvement in TPS, which is one of the objectives of this work.

There are a handful of research works available which address the usage of Blockchain in IoT, mostly as an application use case. In [17] a distributed Blockchain based secure SDN architecture for IoT has been proposed, to enable high-performance availability flow-rule tables in distributed Blockchain. It primarily discusses how IoT communication can benefit from Blockchain technology. In [18], authors propose satellite chains method for improving the transaction rate. Slock [19] explores how to address security, identity, coordination, and privacy across millions of devices by making them autonomous without a middleman. Blockchain-based intelligent transportation system, $B^2ITS$ [20], although not an IoT solution, discusses a seven layers conceptual model using Blockchain, for large scale vehicular networks. Similarly, [21] proposes use of BC in smart grid for secure transactions. A Security framework for integrating Blockchain technology with smart devices has been proposed in [22]. Filament [23] has proposed an open technology stack that enables devices to discover, communicate, and interact with each other autonomously and in a distributed manner. In Provenance [24], a proof of existence and transparency has been implemented in supply chain management through Blockchain.

Work in [25] presents a proposal for ITU to build a decentralized framework for Blockchain of things. This is an ongoing study, which highlights scalability, interoperability, and distributed ledger as high level requirements. In order for different consensus mechanisms to work, the IoT network as a whole should be able to implement Blockchain solution efficiently. It is important to note that most of the existing research related to IoT and Blockchain integration does not focus on scalability of ledger or improving the transaction rate. The prime motivation of this work is to create a framework, which can be implemented in IoT networks (irrespective of application scenario), and is scalable with respect to the number of IoT devices.

## III. Scalable Blockchain Framework for IoT

The main objective of the framework is to enable Blockchain scalability in terms of ledger size and transaction rate. The overall goal is to save BC ledger from extra burden of millions of local transactions within enterprises or home networks.

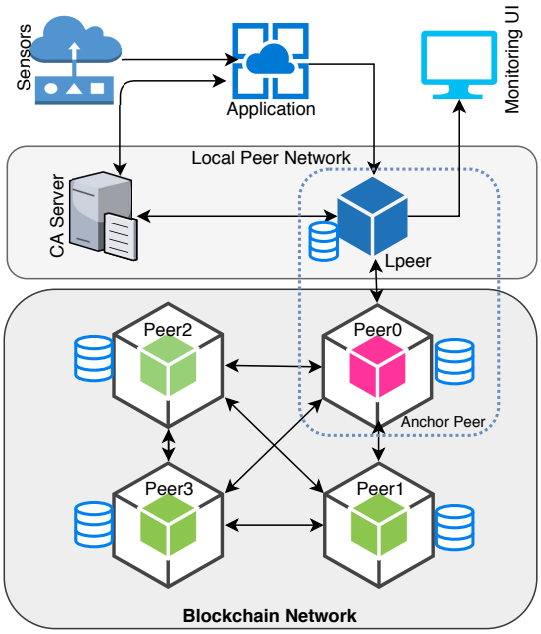



Fig. 2: Local peer based network model.

### A. Working Principle and Network Model

The fundamental principle of the proposed scheme is that IoT devices should not be connected to a peer directly. By using an intermediate entity between devices and BC peers, the flow of transactions can be controlled. Moreover, all IoT devices must have an association with an organization. This organization implements a local peer network, that segregates transactions which remain within the network, from the ones which have to be processed by global Blockchain. In the proposed framework, this principle is implemented by dividing the network into three parts, as shown in Fig. 2.

*1) Application:* In global IoT infrastructure, billions of sensors, actuators, and smart IoT devices are interconnected via Internet. These devices are controlled and managed by applications, either in the form of firmware or gateway applications. In this model, we consider the application as a representation of a *thing*. It is responsible to provide front-end interaction support with other network elements. As the format and structure of data generated by different applications is different, we assume that each application is capable of using a Blockchain standard development kit (SDK) to create transactions and smart contracts.

*2) Local Peer (Lpeer) Network:* In order for an organization to use Blockchain, it must have at least one peer as part of the global Blockchain network. Hence, we propose a Local peer network, which comprises of a Certification Authority (CA) and a Local Peer (Lpeer). The Lpeer network is implemented at organizational level, and groups a number of devices based on their application scenario. CA provides with authentication and registration of devices/users (device certification and associated smart contracts) in the network. Lpeer works as a localized peer for the organization, and provides interaction with an Anchor Peer in the global Blockchain network. It is important to note that Lpeer and Anchor Peer are separate entities with different functionality. This structure indirectly increases the transaction rate of peers in Blockchain, and directly improves the ledger scalability of anchor peer.

*3) Blockchain Network:* In the global Blockchain network, peers are interconnected and every peer maintains its own ledger and holds related smart contracts (chaincode). In the proposed structure, the Lpeer communicates with a corresponding peer in core (referred to as anchor peer) on behalf of applications (already authenticated by CA). The working of the core Blockchain remains the same. The anchor peer is responsible for communicating with other peers (which may be working as anchor peers for other organizations). As shown in Fig. 2, Peer0 acts as an anchor peer and communicates with Lpeer. Inter-organizational transactions are done through the core network of anchor peers. It is also possible that the core Blockchain peer are directly connected to clients, and work as generic peers.

### B. Local Peer Network: Design Details

Implementation details of local peer and associated system elements is shown in Fig. 3. As described earlier, a single application instance represents an individual device which can generate transactions. Anchor peer is part of the global Blockchain network. The rest of figure represents the complete local peer network. Lpeer network is implemented at organizational level. Although IoT devices are capable of D2D communication, but they cannot function without an association to some organization. For example, parking meters installed in a community can be IoT devices (and can generate transactions), but without a parking services organization to maintain, collect data, authenticate, etc., they will not function. Keeping this example in perspective, the local peer network has the following elements.

*1) Certificate Authority (CA):* Certificate Authority Server is a fully trusted entity and supports a range of credential certification architectures. It is an integral part of the architecture, as there is no other entity in a Blockchain network to provide certificates, signatures, and keys. It generates all certificates for administrator including certificates for user registration. All user applications connect to CA for obtaining their encryption keys and signatures. Furthermore, it provides credential validation, signature generation & verification, and TLS-secured connections between all components of blockchain. However, it is not responsible for access control directly. Devices/applications have to implement that solution, and many request it for signature verification only.

*2) Local Peer (Lpeer):* Local Peer only works for the organizational IoT devices. In the proposed framework, we divide the local peer into $\text{Lpeer}^0$, $\text{Lpeer}^1$, to $\text{Lpeer}^N$, where $\text{Lpeer}^0$ is the main instance while all others are secondary instances geographically distributed to remove a single point of failure. Moreover, application scenarios where more than one peer is desired for consensus in local transactions, secondary Lpeers can participate. Selective secondary Lpeers also maintain a replica of ledger. All devices must register with $\text{Lpeer}^0$. It authenticates each device through CA and maintains an active list of users, their credentials, and smart contracts. $\text{Lpeer}^0$ is

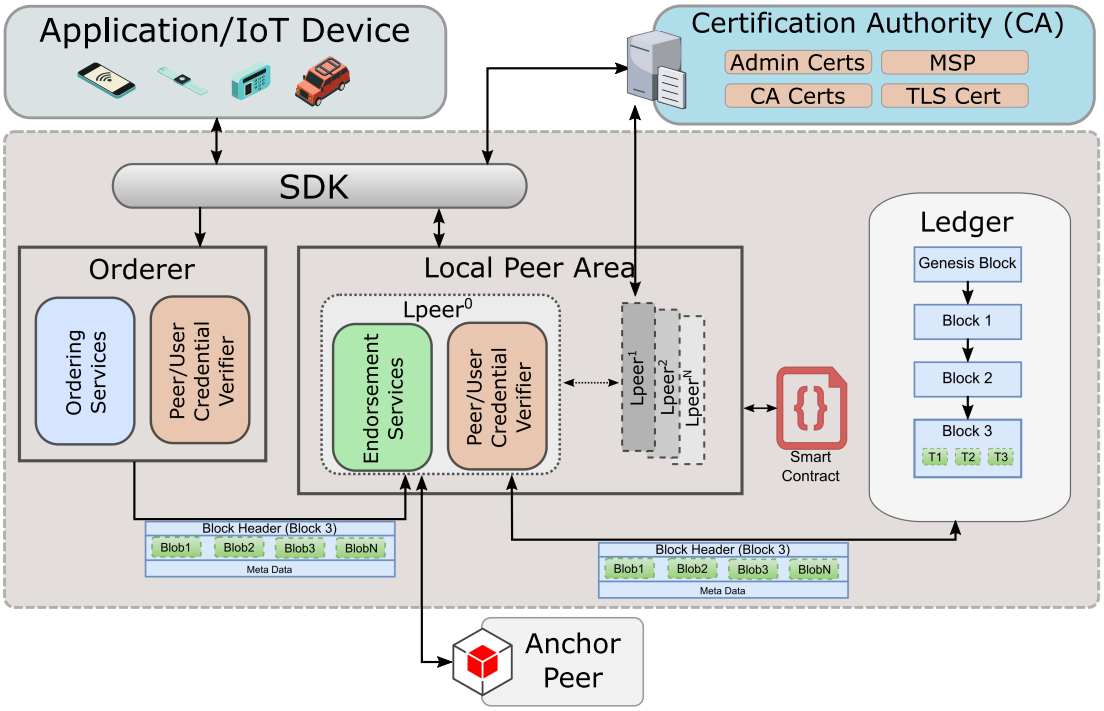



Fig. 3: Local Peer Network: Subcomponent Interaction

the only instance which is allowed to read/write blocks into the ledger. Moreover it is also responsible for interacting with anchor peer for inter-organizational transactions.

*3) Orderer & Ordering Services:* Orderer provides ordering services and may handle transactions of several Lpeers (if any). The main responsibility of orderer is to receive transactions from the different applications/devices and fit into a block according to the ordering algorithm's batch instructions. It stores a copy of all CA generated certificates and signatures. When any user invokes a transaction, orderer uses its own certificates & signature for validation. It is important to note that orderer is not a miner. Orderer's responsibilities are specific to collecting transaction and creating blocks. Miner usually works in crypto currency blockchains for establishing reliability through proof-of-work consensus algorithms.

*4) Ledger:* It is a tamper-resistant and serialized record of all transactions. Transactions are a result of chaincode invocations submitted by participating parties within the organization. Ledger is associated to $\text{Lpeer}^0$, who has permissions to read/write to it. Secondary peers may maintain an auto-updated replica of the ledger, which is used only if $\text{Lpeer}^0$ is out of service. In this work, both Lpeer and BC use state database for maintaining logs for verification of successful transactions.

*5) SDK:* In Blockchain, SDK serves as a shim package to prepare the transaction proposal into well defined format by using user's cryptographic credentials. In this proposal, we use the SDK to format the data and transactional information from a variety of applications into a standard format.

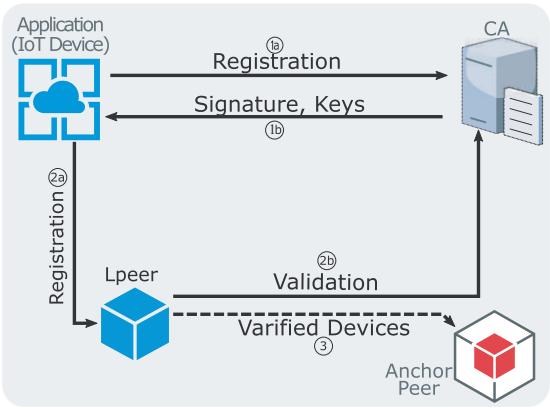
*6) Smart Contract:* Smart contract is a digital contract which defines the terms and conditions of a transaction between two devices. It is implemented in the form of chaincode based on business model and asset definitions. This work uses the smart contract as they are defined for global Blockchain networks [26].

### C. Transaction Structure and Processing

*1) IoT Device Registration:* In order to become part of the system, each IoT device must be registered with CA and $\text{Lpeer}^0$, in the sequence shown in Fig. 4a. The registration with CA is the first step, where CA will provide the devices with a unique signature, and encryption key pairs.

The CA is responsible to create different certificates (i.e. TLS CA, eCert, etc.), signature, pubic-private keys, and gives them to the device. Using these, it then registers with $\text{Lpeer}^0$, which verifies the identity of requester from CA. During the verification stage, $\text{Lpeer}^0$ stores all credentials of devices (i.e. TLS, CA certificates and signatures) for future use of verification. This process ensures that only authorized IoT device can be part of the local Blockchain network. Furthermore, as the IoT device will also be part of the global Blockchain network, where $\text{Lpeer}^0$ registers devices with the anchor peer. The local registration process follows Algo. 1 to complete the device registration process. $CA$ primarily generates and stores the signature and cryptographic materials (e.g. certificates) as response to device registration request (only if accepted). When the device $d_i$ is linked with $Lpeer^0$, then registration ID for the device becomes $Peer_{id}.Device_{id}$, which is used for later transactions.

*2) Transaction Processing:* For every transaction received by $\text{Lpeer}^0$, it verifies if the transaction is coming from a valid user. If the devices is registered, then Algo 2 is used for verification.




(a)

**Transaction from Application to Endorsing Peers**

| Client ID | Chaincode ID | Trans. Payload | Time stamp | Client Sig. | D Client ID |
|---|---|---|---|---|---|
| 4ce32c8... | mycc | 204c33043d6... | 8752475544 | 0xc421bfc420 | 2de1f81.. |
| 2abc3ef.. | mycc1 | 3045ea3c52.. | 9924755565 | 0x24be34c420 | 2de1f81.. |

**Feed Back from Endorser to Applications**

| Ep ID | Trans. ID | Chaincode ID | Tr. Cont. Blob | Read set | Write set |
|---|---|---|---|---|---|
| 8d8a23b... | 204c3367b3... | mycc | 204c33043d... | 1 | 1 |
| 7ebc321e... | 125be6183... | mycc | 354c33043d... | 1 | 1 |

**Application to Orderer**

| Seq. No. | Previous Hash | Blob |
|---|---|---|
| 1 | 46ee8a063.... | 204c33043d... |
| 2 | 2cd43d22... | 3215ef644 |

(b)

Fig. 4: (a) IoT device registration process, (b) Generic transaction format of Hyperledger.

**Algorithm 1:** Device registration

**Input :** Device id $d_i = (d_1, d_2, d_3 ..... d_n)$
**Output:** $Peer_{id}.Device_{id}$

Set $device_{id} \leftarrow d_i$
Set $Peer_{id} \leftarrow Lpeer^0$
Request sign & certificates of $d_i \rightarrow CA$
**if** $d_i(sign \ \& \ certificates)$ **then**
  $d_i \rightarrow Lpeer^0(sign(d_i), d_i)$
  **if** $Lpeer^0(sign(d_i))$ **then**
    **Comment:** Device $d_i$ registered with Lpeer
    *return* $Peer_{id}.Device_{id} = Lpeer^0.d_i$
  **else**
    $d_i$ sign or certificates is not valid
  **end**
**else**
  Request Cancel
**end**

Transaction proposal as message $TP_{i,j}(M_{i,j})$ is forwarded from device $(d_i)$ to Lpeer $P_{Lpeer}$, where $i = 1, 2, 3.....n$ express the devices and $j = 1, 2, 3.....n$ represents messages from each $d_i$. $sk$ and $pk$ denote private and public keys of device $d_i$ respectively. Similarly, $Lp_{pk}$ denotes public key and $Lp_{sk}$ the private key of Lpeer. Transaction proposal is created as $TP_{i,j} = Encrypt(Lp_{pk})[Sign_{d_i}, Sign_{Ad}(d_i), Hash(M_{i,j})]$ where $Ad$ denotes admin of connected device in peer. Peer verifies the transaction proposal messages $TP_{i,j}$ and decrypts by the private key. After decrypting, it verifies the signature of device $Sign(d_i)$ and Lpeer admin $Sign_{Ad}(d_i)$ including all certificates. When all verifications are positive, Lpeer signs and sends a positive response to the source application.

Signature is the key security element of this registration processing mechanism. For generating and verifying the digital signature, it follows the following processes. Assume the $d_i$ will sign a message $M_{i,j}$. $d_i$ creates one private key integer $d_{pi} \in (1, n-1)$ and one public key $Q = d_{pi} \times G$. Here $G$ is a generator of the elliptic curve with large prime order $n$. $d_i$ choose any a random integer $k \in (1, n-1)$ and calculate $e = Hash(M_{i,j})$ and calculates the curve point as $(x_1, y_1) = k \times G$ where left most bit of $e$ is $z$. Calculate $r = x_1 \ mod \ n$ and $s = k^{-1}(z + rd_{pi}) \ mod \ n$ where $r \neq 0 \ and \ s \neq 0$. Finally the signature is the pair $(r, s)$.

Lpeer verifies and accepts the signature initially as valid, if $(r, s) \in (1, n-1)$, otherwise rejects. For validating, calculate $Hash(m)$ with $SHA-1$ algorithm and convert its result to an integer $e$ and $w = s^{-1} \ mod \ n$. Lpeer's signature is acknowledged as $Lp^{ack}_{sign} = (x_1, y_1) = u_1G + u_2Q$ where $u_1 = z \times w \ mod \ n$ and $u_2 = rw$. Here $z$ is the left most bit of $e$. The signature of $d_i$ is accepted as valid if $r \equiv x_1 (mod \ n)$. The signature $(r, s)$ must be invalid if $(r, s) = 0$.

*3) Transaction Flow:* In Blockchain based IoT, there are two types of transaction from device perspective. 1) Transactions among devices registered with same Lpeer, and 2) Transactions among devices registered with different Lpeers. The overall transaction flow is illustrated in Fig. 5. The proposed framework is independent of transaction formats. Generic formats of Hyperledger (with example) are shown in Fig. 4b for reference.

In the first scenario, where both devices are registered with same Lpeer, intra-organizational transaction processing takes place. The transaction request from source is forwarded to

**Algorithm 2:** Device authorization/verification

**Input :** Requested $(Peer_{id}.d_{id})$
**Output:** True or False

**if** $[Peer_{id}.d_{id}]_{prefix} \equiv Lpeer$ **then**
  **if** $sign(Lpeer_{admin}) \ and \ sign(d_i) \ is \ verified$ **then**
    *return* True
  **else**
    *return* False
  **end**
**else**
  *return* False
**end**

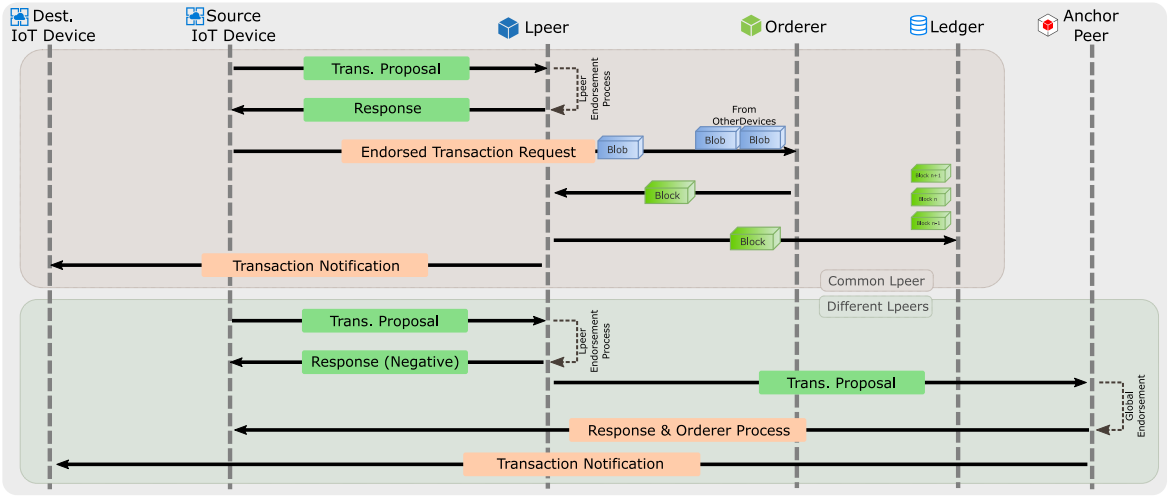



Fig. 5: Transaction information flow.

Lpeer$^0$ through SDK along with the chaincode ID, where it is authenticated and evaluated. Transaction result is produced by executing chaincode and response values are set (as read-/write value). These values and signature of endorsing peers are sent as proposal response, back to the application. The application verifies the signatures of endorser and proposal responses and forwards the transaction to ordering service to be added to the next available block. The orderer may receive endorsed transactions from different applications concurrently. All concurrent transactions are added to a block according to the orderer environment configuration variables. Orderer sends the block to the committing peers (Lpeers) to be added in their ledger, and a notification message to the application.

In the second scenario, as the devices are not registered with same Lpeer, hence anchor peer is used for global endorsement. The inter-organizational transactions will follow generic core BC network endorsement solution. Once the local peer determines that it should not endorse such a transaction request, it notifies the source application, and forwards the request to anchor peer. Complete endorsement process is followed (similar to earlier case, but at global Blockchain level), and result is returned directly to the application. A special scenario can occur where one of the devices does not conform to the proposed Lpeer structure, or both devices do not conform. In either case, as the endorsement will be done at the global Blockchain level, no special processing is required by Lpeer.

*4) Design Implications:* Before delving into the implementation and evaluation details, here we elaborate on some of the technical implications of the design.

- Transaction requests are divided between anchor peer and local peer, which significantly reduces the load on peers (in global Blockchain) both in terms of endorsement time required and ledger size. Moreover, CA and secondary Lpeers provide an additional safety net.
- Compared to traditional mechanism the ledger is now divided. This division is only for the transactions associated to the Lpeer organization. As far as global Blockchain peer network is concerned, anchor peer has the complete ledger for transactions processed by the global network. Hence, the division does not affect the working or trust among peers in BC.
- Ledger maintained by Lpeer is not private, but rather permissioned. If an outside entity needs to obtain a specific organization's internal transactions, they can access them through the anchor peer. Only functional logic needs to be implemented in peers to enable it.
- Lpeer$^0$ is the only Lpeer which is allowed to write into the ledger. Lpeer$^N$ can only be part of endorsement process. The purpose of having multiple instances is to enable scalability inside organizations, and have backup in case of Lpeer$^0$ failure. All Lpeer instances cannot be allowed to write to a single ledger, as it will create synchronization problems. Moreover, all of them should not have individual ledgers, otherwise the problem of global blockchain scalability will trickle down to organizational level, defeating the main purpose.
- Unauthorized transactions might be performed in two ways: 1) Adding illegal peers or device in the network, or 2) A legitimate device becoming rouge. For the first case, we use a CA within the organization to tightly monitor the addition of users (Algo. 1), where as the addition of Lpeer is a controlled process by the network administrators. Compromising an Lpeer does not work for the benefit of hacker, as the BC principle of consensus involves multiple peers. Similarly, a legitimate user going rogue, also cannot issue illegal transactions as Lpeer consensus will not approve them.

## IV. Implementation and Evaluation

The objective of this research is to enable Blockchain scalability in Internet of Things. This is primarily done through better transaction execution rate and smaller ledger size at the peers. It is very important to understand that, Blockchain is an evolving technology, and so is IoT. The architectures of platforms available are rapidly changing, and will affect the evaluations. In the following subsection we provide all

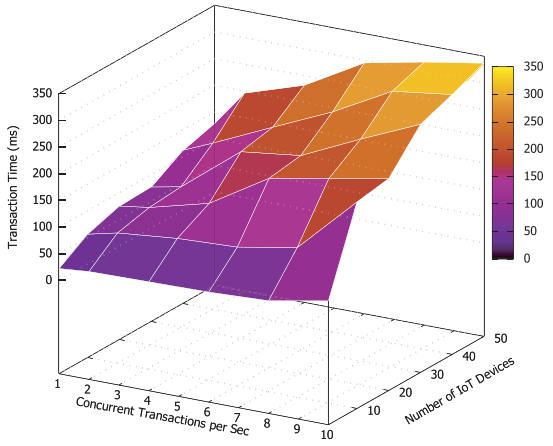



Fig. 6: Transaction scalability.

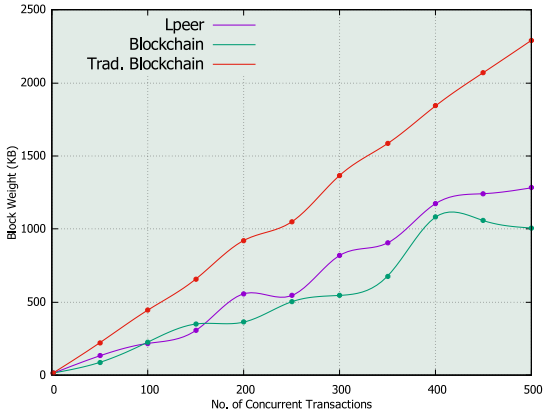



Fig. 7: Block weight for concurrent application transactions.

possible details of the testbed used to evaluate the proposed framework. It is necessary to discuss the technical details, as minor configuration changes drastically change the performance in BC.

### A. Hyperledger Testbed Setup

We have tested our proposal using Hyperledger Fabric (v1.0.2). The testbed comprises of two machines with following specifications for emulating the topology: a) 2.7 GHz, Intel i-7, 16 GB 1600 MHz DDR3, and b) 3.0 GHz, Intel i-5, 8 GB 1600 MHz DDR3. The first machine emulates the Lpeer Network, while the second emulates the global Blockchain. Ubuntu container based virtualization technique has been used to run the peers. $Node-red$ based application generates continuous transactions in JSON payload messages to local peer via smart contract. The local peer has one orderer and one peer organization associated with it, with one channel (lchannel) and kafka ordering service. For the global Blockchain network, we have created four peers, with two controlling organizations. There is an ordering service which connects to *kafka-zookeeper*. *Configtxgen* tool has been used for the creation of genesis block (first block of Blockchain without any hash for previous block), one channel (broadcasting transaction to the orderer), and one anchor peer associated to the Lpeer. We also point to the location of Membership Service Provider (MSP) path for every member. Orderer defines configuration or structure of the transaction as, number of transactions per block, maximum size of a block, and maximum time to wait for transactions. In this setup for Lpeer, we adjust these values according to experimental requirements (described later). We have used *kafka* consensus algorithm for Blockchain network.

### B. TPS Observations

With billions of IoT devices generating transactions, the rate of processing transactions has to be extremely fast at the peers. By design, the proposed framework isolates the organization based local transactions, hence reducing the number of requests per peer. From experimentation we observed, that for Lpeer network, time for generating each certificate for user requires $\approx$ $3ms$. Although, for completion of a transaction (proposal to acknowledgment) it required $\approx$ $3s$. Close observations reviled that most of the time is consumed in generating transaction proposal at application level. By increasing the number of concurrent transactions issued by applications, the average TPS (proposal to acknowledgment) drops below 1 sec. In comparison, it takes approximately 1 hour in bitcoin network to complete the transaction.

Fig. 6 shows overall transactions (Tx) completion time against increasing number of IoT devices and concurrent transactions. The batch timeout (block closing) is set to 1 sec for TPS evaluation in this scenario. It is important to understand that an IoT device with 5Tx per second makes 5 concurrent Tx. Similarly, 50 IoT devices make 500 concurrent Tx (10Tx per device per second). As mentioned in section III, Lpeer is the sole authority to separate the transaction path.

According to Fig. 6, for single IoT device, $1Tx$ takes $\approx 20ms$ while for 10 transactions requires $\approx 57ms$. Likewise, from 20 IoT devices, 20 to 200 concurrent transactions takes $\approx 75-222ms$ respectively. With 20 times rise of concurrent transactions, the time requirement increases at a very slow pace. Finally, with $30-50$ devices, network handles 30 $to$ 500 concurrent transactions where it takes lowest $\approx 77ms$ and maximum $\approx 337ms$. We observe that the increase of transaction completion is not sharp. The main reason is the Lpeer structure which efficiently handles the volumes of Tx.

Once the transaction proposal has been generated, the average transaction time in Lpeer network is $\approx 1-10ms$, whereas in the global Blockchain network it is $\approx 2-56ms$. We closely observe that transaction completion time (only BC and Lpeer level excluding application level) is $\approx 1-56ms$.

TPS is highly dependent on proper setting of environment variables, which are dictated by the volume of transactions, size of individual transaction, log level, etc. During the experimentation it was observed that the processing time is impacted even by the trivial misconfiguration, example of which is, by leaving the log visibility mode *on* for Hyperledger, the average processing time increased from 20ms to 32ms.

### C. Block Weight & Ledger Scalability Analysis

Blockchain maintains a continuously growing list of ordered transactions called blocks in the ledger. Each block number is unique and is assigned sequentially starting from zero. Each block is linked with the previous block except genesis block. Every block has three sections: header, transaction information, and metadata.

*1) Individual Block Weight:* The weight of the block refers to the memory required to store it. This is different than the size of a block, which refers to the number of transactions enclosed in a block. Furthermore, the contents of a block are stored as a string (i.e. 4-byte integer may take more memory when converted). In our analysis, we have observed the actual weight of the ledger stored on the peer nodes, in order to obtain the real memory requirement. After a number of experiments, on average the block weight observed was $\approx 4.6KB$ with a single transaction, endorsed by a single peer. This value is highly dependent on the weight of transaction itself, number of transactions in a block, and the number of endorsing peers for each transaction. As each endorser's signature is added to the transaction, hence the weight grows with the number of endorsing peers.

*2) Block Weight vs. Concurrent Transactions:* As discussed earlier, IoT devices will generate transactions at a much higher rate as compared to current cryptocurrency transactions. Hence we analyze the effect on weight of block against increasing number of concurrent transactions from applications. The batch timeout is 50s (block closing time). This value is too high for production level systems, but it allows us to study the higher number of transactions in a block. The maximum messages per batch is 2K and maximum weight per batch is 100MB. The preferred maximum number of bytes per message is set to 512KB.

Fig. 7 shows the graph for block weight against number of transactions issued by applications. The red line represents traditional Blockchain architecture, and it almost grows sequentially with the increase. The other two lines represent the block weights in the proposed framework generated by Lpeer and anchor peer. It is important to note that the probability of transaction to be intra-organizational is 0.7. This probability is realistic, as a parking meter is more likely to generate a transaction with other parking service devices, and less likely to exchange data with a temperature sensor. The block weight increases for Lpeer and anchor peer, but not at the rate of traditional BC architecture. Based on transacting parties, the transactions are divided among Lpeer and anchor, hence reducing the block weights. Moreover the block weight for Lpeer is relatively higher than that of anchor due to significantly more transactions per block. Although the anchor peer gets less transactions to process, but validation by four peers adds to the block weights. In production environments more peers will be present hence increasing the weight of block for anchor peer.

*3) Scaling of Ledger:* Fig. 8 represents scalability of ledger in both frameworks. As blocks are added in sequential order, hence block height refers to the number of blocks added to ledger. The memory shown is cumulative. Block 0 is the genesis block and takes approximately 12KB in most cases.

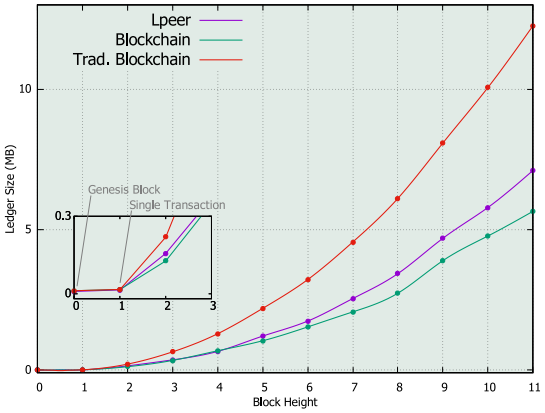



Fig. 8: Scalability of ledger memory.

Block 1 holds only instantiated data and contains 1 transaction. In traditional Blockchain network, from block 2 to block 11 memory size increase is sharp. On the other hand, for the same number of transaction and blocks, proposed framework BC ledger requires significantly less memory resources. Furthermore, as all the peers in BC network maintain copies of the same ledger, hence the overall consumption is proportional to number of peers. With Lpeer, as there is only one ledger per organization, it becomes more scalable across different IoT domains.

## V. Conclusion

Rapid spread of IoT devices has prompted a renewed search for security and privacy solution for data and users. Similarly, Blockchain technology has emerged as a candidate for secure transaction based communications. In this article, we show that IoT and Blockchain cannot be integrated, unless scalability issues are addressed. The proposed framework creates a local peer network in order to allow Blockchain ledger to scale across all peers. The results obtained from implementation testbed show that significant improvement in TPS and ledger weight can be achieved. This will allow better scalability of large scale business transactions in IoT, and will address the memory requirement issue to store the blocks. The current implementation and evaluation has been done in part an virtual machines, where the application is implemented in $Node{-}red$. As future work, we plan to implement the solution in real world situation, and evaluate using real time transactional data. Moreover, the solution proposed focuses only an scalability and resource conservation, and uses existing transaction structures without any optimization to them. It will be an interesting direction to have either a unified transaction structure which is optimized for different types of business chains, or specialized structures but with interoperability among multiple chains.

## References

[1] J. Huang, C. C. Xing, S. Y. Shin, F. Hou, and C. H. Hsu, "Optimizing m2m communications and quality of services in the iot for sustainable smart cities," *IEEE Transactions on Sustainable Computing*, vol. 3, no. 1, pp. 4–15, Jan 2018.

[2] Amy Nordrum, "Popular Internet of Things Forecast of 50 Billion Devices by 2020 Is Outdated," Accessed: 2018-05-31. [Online]. Available: https://spectrum.ieee.org/tech-talk/telecom/internet/popular-internet-of-things-forecast-of-50-billion-devices-by-2020-is-outdated

[3] E. Luo, M. Z. A. Bhuiyan, G. Wang, M. A. Rahman, J. Wu, and M. Atiquzzaman, "Privacyprotector: Privacy-protected patient data collection in iot-based healthcare systems," *IEEE Communications Magazine*, vol. 56, no. 2, pp. 163–168, Feb 2018.

[4] J. Pan and J. McElhannon, "Future edge cloud and edge computing for internet of things applications," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 439–449, Feb 2018.

[5] H. Sun, Z. Zhang, R. Q. Hu, and Y. Qian, "Challenges and enabling technologies in 5g wearable communications," *CoRR*, vol. abs/1708.05410, 2017.

[6] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," Accessed: 2018-05-31. [Online]. Available: http://bitcoin.org/bitcoin.pdf

[7] A. Lielacher. How many people use bitcoin? updated for 2018. Accessed: 2018-05-30. [Online]. Available: https://www.bitcoinmarketjournal.com/how-many-people-use-bitcoin/

[8] M. Beck. Into the ether with ethereum class. Accessed: 2018-05-31. [Online]. Available: https://grayscale.co/wp-content/uploads/2018/03/Grayscale-Ethereum-Classic-Investment-Thesis-August-2017.pdf

[9] A. B. David Schwartz, Noah Youngs, "The ripple protocol consensus algorithm," Accessed: 2018-05-31. [Online]. Available: https://ripple.com/files/ripple_consensus_whitepaper.pdf

[10] Hyperledger: Hyperledger business blockchain technology. Accessed: 2018-05-31. [Online]. Available: https://www.hyperledger.org/projects

[11] E. Androulaki, A. Barger, and e. Bortnikov, "Hyperledger fabric: A distributed operating system for permissioned blockchains," in *Proceedings of Thirteenth EuroSys Conference*. ACM, 2018, pp. 30:1–30:15.

[12] C. Cachin and M. Vukolic, "Blockchain consensus protocols in the wild," *CoRR*, vol. abs/1707.01873, 2017. [Online]. Available: http://arxiv.org/abs/1707.01873

[13] C. Xu, C. Gao, Z. Zhou, Z. Chang, and Y. Jia, "Social network-based content delivery in device-to-device underlay cellular networks using matching theory," *IEEE Access*, vol. 5, pp. 924–937, 2017.

[14] M. Vukolić, "Rethinking permissioned blockchains," in *ACM Workshop on Blockchain, Cryptocurrencies and Contracts*. ACM, 2017, pp. 3–7.

[15] O. Novo, "Blockchain meets iot: An architecture for scalable access management in iot," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 1184–1195, April 2018.

[16] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, "LSB: A lightweight scalable blockchain for iot security and privacy," *CoRR*, vol. abs/1712.02969, 2017.

[17] P. K. Sharma, S. Singh, Y. S. Jeong, and J. H. Park, "Distblocknet: A distributed blockchains-based secure sdn architecture for iot networks," *IEEE Communications Magazine*, vol. 55, no. 9, pp. 78–85, 2017.

[18] W. Li, A. Sforzin, S. Fedorov, and G. O. Karame, "Towards scalable and private industrial blockchains," in *ACM Workshop on Blockchain, Cryptocurrencies and Contracts*. ACM, 2017, pp. 9–14.

[19] Slock: Secured lock. Accessed: 2018-05-31. [Online]. Available: https://slock.it/technology.html

[20] Y. Yuan and F.-Y. Wang, "Towards blockchain-based intelligent transportation systems," *Proceedings of Eighteenth International Conference on Intelligent Transportation Systems*, pp. 2663–2668, 2016.

[21] F. Gao, L. Zhu, M. Shen, K. Sharif, Z. Wan, and K. Ren, "A blockchain-based privacy-preserving payment mechanism for vehicle-to-grid networks," *IEEE Network*, pp. 1–9, 2018.

[22] K. Biswas and V. Muthukkumarasamy, "Securing smart cities using blockchain technology," in *Proceedings of International Conference on High Performance Computing and Communications*, Dec 2016, pp. 1392–1393.

[23] Filament, "Security Overview: Networking Devices at the Edge of Risk," White Paper, 2017, Accessed: 2018-05-31. [Online]. Available: https://filament.com/assets/downloads/Filament%20Security.pdf

[24] Provenance, "Blockchain: the solution for transparency in product supply chains," White Paper, 2015, Accessed: 2018-05-31. [Online]. Available: https://www.provenance.org/whitepaper

[25] X. Jia, R. A. Fathy, Z. Huang, S. Luo, C. Ma, J. Peng, A. A. Hassan, and H. Wan, "Framework of blockchain of things as decentralized service platform," Version: ITU-T SG 20-TD779 Proposal, 2018 Accessed: 2018-09-07. [Online]. Available: https://www.itu.int/md/T17-SG20-180506-TD-GEN-0779/en

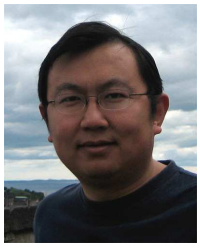
[26] I. O. Kennedy, C. K. Lin, and V. Venkateswaran, "A cross layer design and evaluation of ieee 802.15.4 network with an enhanced sensor gateway: Injecting hierarchy into wireless sensor networks," in *Proceedings of IEEE International Conference on Communications*, June 2013, pp. 1694–1699.


**Sujit Biswas** (GS'17) is enrolled as PhD fellow in Beijing Institute of Technology, China. He received his M.Sc. degree in Computer Engineering from Northwestern Polytechnical University, China in 2015. He is also an Assistant Professor with Computer Science and Engineering department, Faridpur Engineering College, University of Dhaka, Bangladesh. His basic research interest is in IoT, Blockchain, Mobile computing security and privacy.

**Kashif Sharif** (M'08) is an Associate Professor (Research) at School of Computer Science and Technology, Beijing Institute of Technology, China. He received his PhD degree in Computing and Informatics from University of North Carolina at Charlotte, in 2012, and MS degree in Information Technology from National University of Sciences & Technology, Pakistan, in 2004. His research interests include wireless & sensor networks, network simulation systems, software defined networks, data center networking, information centric networking, and blockchain technologies. He is a member of the IEEE.

**Fan Li** received the PhD degree in computer science from the University of North Carolina at Charlotte in 2008, MEng degree in electrical engineering from the University of Delaware in 2004, MEng and BEng degrees in communications and information system from Huazhong University of Science and Technology, China in 2001 and 1998, respectively. She is currently a professor at School of Computer Science in Beijing Institute of Technology, China. Her current research focuses on wireless networks, ad hoc and sensor networks, and mobile computing. Her papers won Best Paper Awards from IEEE MASS (2013), IEEE IPCCC (2013), ACM MobiHoc (2014), and Tsinghua Science and Technology (2015). She is a member of ACM and IEEE.

**Boubakr Nour** (GS'17) is pursuing his Ph.D. in Computer Science & Technology at Beijing Institute of Technology, Beijing, China. Previously, he received both M.Sc. and B.Sc. degrees with distinction in Computer Science from Djillali Liabes University, Sidi Bel Abbes, Algeria, in 2016, and 2014 respectively. Also, he has been a visiting researcher at Paris Descartes University, Paris, France, in 2016. In the year 2016-17, he won the Excellent Student Award at the Beijing Institute of Technology. His research interests include Next-Generation Networking and Internet, Information-Centric Networking, Named Data Networks, Mobile/Edge Computing, Internet of Things, and Wireless Sensor Networks.



**Yu Wang** (M'01, SM'10, F'18) is a Professor of Computer Science at the University of North Carolina at Charlotte. He received his Ph.D. degree in Computer Science from Illinois Institute of Technology, his B.Eng. degree and M.Eng. degree in Computer Science from Tsinghua University, China. His research interest includes wireless networks, mobile social networks, smart sensing, mobile crowd sensing, mobile computing, and algorithm design. His research has been continuously supported by federal agencies including US National Science Foundation, US Department of Transportation, and National Natural Science Foundation of China (NSFC). He has published over 150 papers in peer reviewed journals and conferences, with four best paper awards. He is a recipient of Ralph E. Powe Junior Faculty Enhancement Awards from Oak Ridge Associated Universities (2006), Outstanding Faculty Research Award from College of Computing and Informatics at UNC Charlotte (2008), and Overseas Young Scholars Cooperation Research Fund from NSFC (2014). He is a senior member of the ACM and a fellow of the IEEE.