COLUMN: Blue Skies

TrustChain: Establishing Trust in the IoT-based Applications Ecosystem Using Blockchain

Bin Yu
Monash University,
CSIRO Data61

Jarod Wright, Surya Nepal, Liming Zhu
CSIRO Data61

Joseph Liu
Monash University

Rajiv Ranjan
Newcastle University

The Internet of Things (IoT) has already reshaped and transformed our lives in many ways, ranging from how we communicate with people or manage our health to how we drive our cars and manage our homes. With the rapid development of the IoT ecosystem in a wide range of applications, IoT devices and data are going to be traded as commodities in the marketplace in the near future, similar to cloud services or physical objects. Developing such a trading platform has previously been identified as one of the key grand challenges in the integration of IoT and data science. Deployment of such a platform raises concerns about the security and privacy of data and devices since their ownership is hard to trace and manage without a central trusted authority. A central trusted authority is not a viable solution for a fully decentralized and distributed IoT ecosystem with a large number of distributed device vendors and consumers. Blockchain, as a decentralized system, removes the requirement for a trusted third-party by allowing participants to verify data correctness and ensure its immutability. IoT devices can use blockchain to register themselves and organize, store, and share streams of data effectively and reliably. We demonstrate the applicability of blockchain to IoT devices and data management with an aim of providing end-to-end trust for trading. We also give a brief introduction to the topics and challenges for future research toward developing a trustworthy trading platform for IoT ecosystems.

Copublished by the IEEE CS and IEEE ComSoc
2325-6095/18/$33.00 USD ©2018 IEEE

The number of Internet of Things (IoT) devices has already exceeded the world population. With rapid advancement in hardware technologies, these smart devices have been applied in almost every aspect of our daily lives. A large amount of data is generated every second and data science research is actively defining algorithms to process such data to make and enact better decisions for us in our daily activities. For example, wearable smart devices such as smartwatches sense our heartbeat and blood pressure continuously to monitor our health condition; a smart fridge enables us to control the fridge remotely and plan a healthier diet; a smart air conditioner can track our living preferences and adjust the temperature automatically; an autonomous vehicle frees our hands and minds while making our journey safe.

One of the key grand challenges is how we ensure that users trust the IoT ecosystem to make the right decisions and act on them. This involves trusting devices, data and analytics, as previously identified in this column.[1] The focus of this article is to analyze different research and technical issues related to managing trust using blockchain in a fully decentralized IoT ecosystem.

Though these smart devices bring great convenience to us in our daily life, news such as US cell carriers (including AT&T, T-Mobile. and Sprint) selling access to customers' real-time phone location data to a little known company called Securus[2] raises public concerns about the risk of personal data leakage and abuse. Such news prompts a debate on whether these IoT devices are our friends or enemies. Trust is not a one-way street in the IoT ecosystem. Data analysts have concerns about the integrity of the data that data owners provide. At the same time, data owners are concerned about whether data analysts only use their data for its declared purposes. Additionally, data owners care about how to protect their own data (sometime captured by manufacturers) when IoT device ownership changes during its life. For example, what happens to the data of a car owner when an autonomous car is sold or the ownership of a car is changed?

Users find difficulties in enjoying the services provided by these smart devices if they don't meet the high security expectations from them. Some key challenges for building a trustworthy trading platform for IoT devices and data are outlined below:

Lack of trust among participating entities. Trust is hard to achieve among different entities involved in IoT data processing due to the lack of a governance framework. As defined by NIST in its Network-of-Things (NoT)[3] report, which aims to define IoT formally, five key primitives are involved in real IoT applications: sensors (IoT devices for generating data), aggregators (edge, fog or mist infrastructure for aggregating data), communication channels (wired and wireless communication provided by communication service providers), eUtility (SaaS, PaaS, IaaS provided by clouds), and decision triggers (data analysis pipelines, decision making and enacting processes). Each of these primitives is likely to be supported by different service providers. How can they trust to each other? For example, in many cases, IoT device owners would not know who are the data processing entities or cloud service providers. Unless they have a mechanism to trust them to handle the data properly, they cannot use the services they provide to support the five primitives. More seriously, there is no standard agreement among different entities to define the data usage policy and it is hard to supervise the usage of personal data. The reliability and security of all entities providing five primitives is important to establish the trust.

Lack of data supervision and management. In many applications, data collected by IoT devices is mostly maintained and processed by either the device manufacturer or a trusted third-party. For example, consider an IoT application of monitoring chronic patients at home[4] where a patient is monitored for his activities (like exercises) and health (blood pressure, heart beats) using IoT devices. A service provider may share patient data with health data analytics, general practitioners, and related service providers, including cloud data service providers. Patients have limited knowledge about how their data is processed and used. Additionally, when the data generated by IoT devices is transferred from one party to another, there is often no data integrity verification. The tampered data could result in misleading decisions and the source of the fraud is difficult to identify.

Lack of devices' lifecycle management. Every product undergoes a series of phases in its lifecycle: design, sourcing of components, manufacturing, distribution, retail, repair, resale, and so on. For IoT devices, the management of devices and related data are critical because the data generated by devices at different phases should be isolated and well protected. To date, the visibility remains highly siloed and opaque across entities. For instance, patient health data generated by an artificial cardiac pacemaker is fragmented to the manufacturer, doctor, and insurance company. During the device's lifecycle, a patient could change from one doctor to another; in such a scenario, the data accessibility should also be transferred. Currently, the data is fragmented and held by many entities and there is no regulation on auditing the ownership of IoT devices. Similarly, there is no guarantee that the data held by each of them is consistent with others. If the patient is transferred from one doctor to another, there is no mechanism on how the data accessibility is managed.

A lot of effort has been put into resolving the issue of trust among different entities in the IoT application ecosystem. Unfortunately, there is not a single reference scheme that satisfies all stakeholders. The key issue for the traditional solutions is that they all depend on a trusted third-party, which has to be trusted by all stakeholders. Blockchain, as a new data-sharing model, addresses this issue by removing the need for a trusted third-party. It allows all stakeholders to participate in maintaining an immutable ledger in which the data is consistent among all stakeholders. Since the data on the ledger is immutable, we avoid the possibility that any participant tampers with the data by allowing all participants to verify the correctness of the data. In this article, we argue that with the help of the blockchain technology, the management of the IoT device life cycle and the corresponding data privacy can be enhanced in the following ways:

- instead of trusting a third party, IoT devices can exchange data through the blockchain;
- IoT devices and the data generated by IoT devices can be traced to avoid the manipulation of the data by malicious parties;
- different stakeholders can trust the validity and integrity of the data on the chain;
- the communication among different entities can be simplified as they only need to interact with the blockchain to retrieve/upload data;
- the deployment and operation cost of IoT can be reduced through a blockchain since there is no intermediary;
- computation-intensive operations like end-user authentication and access control can be processed on the blockchain instead of IoT devices;
- it is more convenient for the blockchain to maintain device and data ownership; and
- the distributed ledger eliminates a single point of failure within the ecosystem—IoT devices and end users can interact with any blockchain nodes to access the data;

In the following, our aim is to demonstrate the feasibility of a trustworthy trading platform with the above stated capabilities. Before going to a case study of such a platform, we give an overview of the blockchain technology.

BLOCKCHAIN

Trust plays a critical role in information exchanges. It helps different entities deal with each other more effectively and is often a key element in any collaborative system. Traditionally, centralized trusted institutions such as banks or government agencies manage the trust problem. With the help of these centralized institutions, different entities can cooperate with each other with a certain degree of confidence. Blockchain, known as an electronic ledger, tries to replace such centralized institutions by distributing the trust in a decentralized network. In a blockchain system, the ledger is immutable and not held on a single server but among all servers in the network. The openness feature of blockchain allows any participant to modify the ledger under a set of rules dictated by a "consensus protocol." The "consensus protocol" requires the majority of the blockchain participants to agree on the modification of the ledger to ensure the trustworthiness of the blockchain. Once a new consensus is achieved, all participants update their own ledger simultaneously. If any of the participants violates the consensus protocol to propose a new data entry, the network treats that entry as an invalid one.

Practically, transactions are bundled together and submitted to the blockchain as a block. Cryptographic techniques are applied to link all blocks in a deterministic order. The cryptographic algorithm also guarantees that the blocks are immutable, which means that once a block is appended to the chain, it cannot be tampered with.

We take Hyperledger Fabric as an example blockchain platform to illustrate how a smart contract service works in the blockchain. Figure 1 shows how a smart contract is deployed on the blockchain. First, the smart contract administrator needs to compile the smart contract application into a binary code so that it can be executed on the Hyperledger fabric. The administrator then deploys the smart contract on the blockchain. The application receives status notifications when there is any change. For instance, when the smart contract is deployed successfully, the application receives a message saying that the smart contract is now running on the blockchain. Finally, end users can access the service through the interface provided by the blockchain.

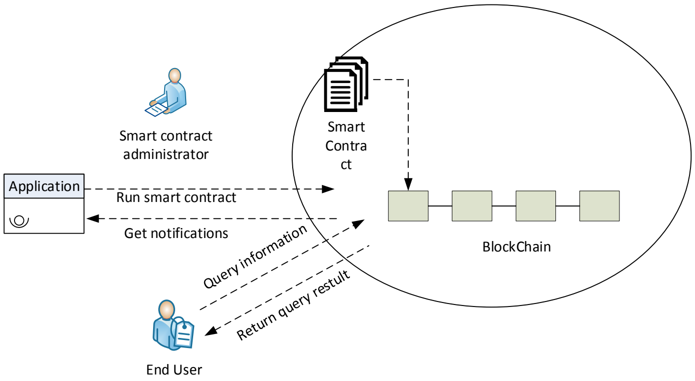

Figure 1. Smart contract on blockchain.

There are typically three different types of consensus protocols. The first is the Proof-of-Work (PoW),[5] adopted by cryptocurrencies like Bitcoin[6] and Zcash.[7] For the PoW protocol, all participants are competing with each other to win the block proposal by solving a specific math puzzle. The first participant to find the solution authorized to propose the transactions in the block and the rest of the participants copy this block to their own chain. A PoW based blockchain can provide an open environment, which allows anyone to join/leave the blockchain freely. However, in this protocol, a high amount of energy is consumed by each participant to solve the math puzzle. As a result, they have a poor throughput. The second type of consensus protocol is based on Byzantine Fault Tolerance (BFT),[8] which is adopted by blockchain systems like Hyperledger Fabric.[9] The size of the network of a BFT-based blockchain is relatively small. As a result, the majority of BFT-based blockchain systems are permissioned blockchains, in which only authorized users can participant in block generation and verification. The last one is the Proof-of-Stake (PoS),[10] which employs a certain number of nodes to generate the blocks on behalf of the whole network. Typical examples of PoS-based blockchain systems are Ouroboros,[11] Neo,[12] and Reddcoin.[13] However, in such systems, rich nodes have more chances to generate blocks. To address such problems, Bitcoin-NG[14] advocates applying a hybrid consensus protocol, which combines the advantages of the PoW and PoS. GHOST,[15] SPECTRE,[16] and MESHcash[17] are recent proposals for increasing the throughput by replacing the underlying chain structure with a tree or a directed acyclic graph (DAG) structure. These protocols still rely on the Nakamoto consensus using PoW. By carefully designing the selection rules between branches of the trees/DAGs, they are able to substantially increase the throughput.[18]

Since the blockchain removes a centralized trusted party and allows a participant to verify the correctness of the data on the blockchain, it is widely applied in two domains—cryptocurrency and the smart contract.[19] The great success of Bitcoin,[6] Litecoin,[20] Zcash,[7] Ripple,[21] and EOS[22] demonstrates the potential market value of the blockchain technology. The smart contract is another application based on blockchain technology. In essence, the smart contract is a service that links different entities together to construct a system to achieve dedicated functions. The approximate turning-complete feature provided by the smart contract allows the majority of existing programs to migrate to the blockchain.

Three key features that the blockchain technology brings to the industry are:

Openness: Trustworthiness is always a key issue for systems that involve multiple parties. All communications between parties are based on a certain kind of trust assumptions. Blockchain technology is so revolutionary that it allows exchanging value directly between parties without them trusting each other. The openness feature allows anyone interested in the system to join and verify the correctness of the data. Because the data on the chain is immutable, it resolves concerns that the data owner might tamper with or modify the data in the future.

Robustness: Denial of service and a single point of failure are common issues for existing centralized systems. If the centralized servers are under attack, the quality of service might be affected and system security could be compromised. However, since every participant holds a copy of the data, and the network size could be large, it is impossible for an adversary to attack the blockchain system by compromising the majority of the distributed blockchain servers.

Cooperation: Blockchain enables a new cooperation pattern among multiple parties in which untrusted parties can exchange data more confidently by hosting the servers locally to construct a blockchain network. For example, take an electronic voting system, when the voting is conducted in a traditional way, all voters and stakeholders should agree on a trusted third party to organize the voting. Blockchain removes this trusted party by allowing all stakeholders to participate in the voting administration. That is, all stakeholders can verify the correctness of the voting result by looking up the data on the blockchain node held by themselves.

There are two paradigms for blockchain resource management: permissioned blockchain and permissionless blockchain. For the permissioned blockchain, a membership service exists that asks all parties who want to contribute in the blockchain maintenance to register with the blockchain system. Hence, only authorized users can access the blockchain. In contrast, for the permissionless blockchain, everyone can access the data on the blockchain and participate in the blockchain management without registering. We make a comparison between permissioned and permissionless systems in Table 1. We briefly describe them below.

Table 1. Comparison between permissioned and permissionless blockchain

	Permissioned Blockchain	**Permissionless Blockchain**
Operational costs	Depends on the redundancy requirements	High (Bitcoin estimate $657,000,000 per year in 2017 at $1000/BTC)
Interoperability	Poor	Excellent
Transaction Throughput	Good	Poor
Data Privacy	Good	Poor
Scalability	Poor	Good
System robustness and resilience	Fair	Good

Permissionless blockchain: The advantages of a permissionless blockchain are: 1) it has an open network to enable anyone to join/quit the protocol freely; 2) the network typically has an incentivizing mechanism to encourage more participants to join the network; and 3) it is suitable for cryptocurrency and applications that do not have strict privacy requirements. However, it consumes a lot of power to maintain the distributed ledger at a large scale for PoW based systems and the trust of the blockchain is hard to achieve for PoS based systems. Furthermore, very limited transaction privacy is preserved since any nodes in the permissionless blockchain can have a copy of all transactions.

Permissioned blockchain: The advantages of a permissioned blockchain are: 1) all blockchain participants are registered and verified by the protocol administrator and as a result, it is easy to identify nodes that do not comply with the protocol; 2) since the public has no access to the blockchain, privacy is preserved; and 3) since the blockchain administrator can control the network size by controlling the number of nodes involved in the blockchain, the permissioned blockchain usually has a high transaction throughput. However, the permissioned blockchain has a number of disadvantages: 1) the public may have low confidence in the correctness of the blockchain because they have no access to the verification of the data on the chain; 2) some stakeholders may collude with each other to make some transactions invalid; 3) participants need to follow a series of strict policies to join/quit the protocol (i.e., a membership server should assign/withdraw its access policy) and 4) some protocols (e.g., PBFT) are based on assumptions that two or three of the total nodes are always online.

CASE STUDY: TRUSTCHAIN – A PLATFORM FOR IOT DEVICE AND DATA TRACKING AND TRADING

In this section, we provide a concrete example of how the blockchain is applied to enhance the trust in a practical scenario to track and trade IoT devices and the corresponding data.

The popularity of wearable devices is increasing and people are regularly upgrading their IoT devices, while few of them understand how to dispose of their out-of-date devices. Trade-in or resale to another customer usually means the potential or accidental transfer of all personal data on the device to the new buyer, resulting in personal data leakage. On the other hand, when a device is put up for sale, the manufacturer needs to trace the ownership of the device to provide a warranty to the correct customer and recall the device if any defects are found. Current faulty airbag recalls on vehicles is a good example—many consumers are not aware that their vehicles have been recalled.

Applying the blockchain technology in the above scenario, we demonstrate how our IoT device and related data tracking and trading system resolves the trustworthiness issue in the IoT ecosystem. Four entities are involved in our system: 1) manufactures, who sell the products to retailers; 2) retailers, who buy the products from manufactures and sell them to customers; 3) customers, who consume the service provided by products; and 4) data analysis companies that buy the personal data for analysis.

For the IoT device and data tracking and trading system, we would like to have the following functions: 1) a manufacturer can trace the status and ownership of the device during its lifecycle; 2) a customer can transfer the ownership of his/her devices to another customer; 3) a customer can share/sell the data generated by his/her IoT devices; and 4) a smart contract that only allows the data owner to sell his/her own data; once the IoT device is sold, he/she cannot access the data generated by that device any longer.

The logical architecture of our tracking and trading system is shown in Figure 2. It demonstrates how cell phones, as IoT devices are transferred from a manufacturer to a retailer, a retailer to a customer, and finally, a customer to another customer. With the help of a smart contract, the manufacturer can transfer the ownership of the cell phone to the retailer. When Alice as a customer purchases this item, a record that corresponds to this purchase is appended to the smart contract, which demonstrates that the cell phone is owned by the customer, Alice. If Alice agrees to sell her cell phone as a used device to Bob, the smart contract can inform the manufacturer that the device is owned by Bob. The manufacturer then provides any remaining warranty service to Bob. Since the device is transferred to Bob, Bob can sell the personal data generated by his cell phone to a data analysis company. At the same time, he has no rights to handle the data generated by the same device previously under the ownership of Alice.

Figure 2 shows the interactions between different entities that are carried out through a smart contract. Since the smart contract is regarded as an independent trusted party, no entity can cheat others or modify the existing data related to this device.

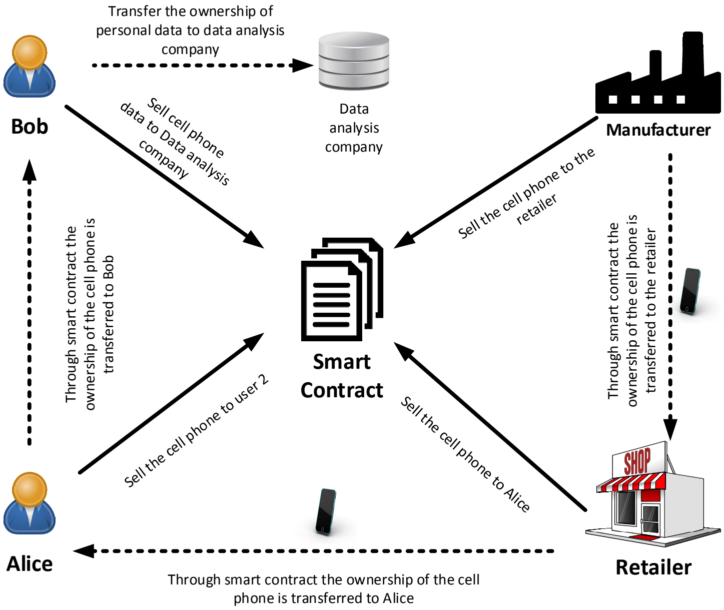
Transfer the ownership of personal data to data analysis company
Bob
Data analysis company
Manufacturer
Sell cell phone data to Data analysis company
Sell the cell phone to the retailer
Through smart contract the ownership of the cell phone is transferred to Bob
Smart Contract
Through smart contract the ownership of the cell phone is transferred to the retailer
Sell the cell phone to user 2
Sell the cell phone to Alice
SHOP
Alice
Retailer
Through smart contract the ownership of the cell phone is transferred to Alice

Figure 2. Device tracking and data trading system.

The above case studies can be implemented in both permissioned and permissionless blockchains. Let us first consider the permissioned blockchain. We employ Hyperledger Fabric as a private blockchain that only allows relevant stakeholders to store IoT devices and the data ownership information. Users who own IoT devices can verify the manufacturer of those devices and sell the data generated by them. Since the public does not own the IoT device, they cannot trace any information related to the device. In practice, to preserve the data privacy, we only allow the manufacturer, retailer, and government consumer affairs office to host the blockchain validation nodes.

The logical structure of Hyperledger Fabric is shown in Figure 3. It consists of the following components.

Client: The client represents the entity that acts on behalf of an end-user. It must connect to a peer to communicate with the blockchain. Clients create and thereby invoke transactions.

Peer: The peer receives the ordered state updates in the form of blocks from the ordering service and maintains the state and the ledger. The peer nodes are held by different stakeholders to ensure that the data on the blockchain are verified by all stakeholders to avoid any party tampering with or creating an incorrect block on the chain.

Ordering service: This service provides a communication channel to clients and peers, and offers a broadcast service for messages containing transactions. The channel outputs the same messages to all connected peers in the same logical order.

Certificate Authority (CA) server: The server is responsible for creating user/server certificates and verifying servers' validity in the network. Peer nodes in the blockchain network also ask the CA server to verify the identity of peer nodes.

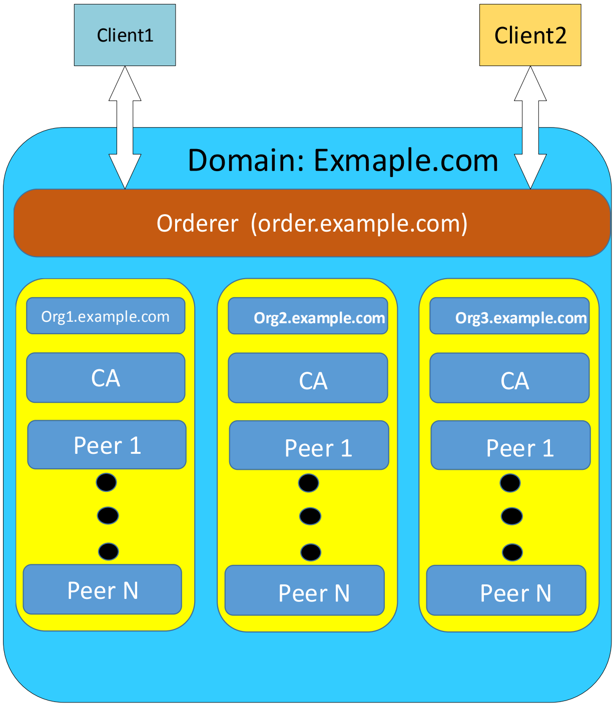

Figure 3. Hyperledger Fabric structure.[4]

SMART CONTRACT LOGIC

In this section, we explain the portal and the interfaces of different entities in our system to illustrate the logic of our smart contract in Figure 4.

The smart contract runs atop the Ethereum blockchain as a form of distributed computation. When the contract is interacted with, either by a retailer or a manufacturer selling/buying a device on the blockchain or by an end user seeking to buy/modify a device, they call one of the functions outlined below. Both inputs and operations of these functions are then computed across the whole blockchain. An individual who invokes the function pays for all nodes in the network to perform that function through the use of “gas.” Gas is the execution fee that scales according to the amount of computational power needed to perform the invoked function by all nodes doing the computation. This execution fee is the incentive for nodes in the system to actually perform the necessary computations to ensure the contract is obliged by the blockchain. As the computation is distributed and performed by all nodes in the Ethereum blockchain, the need for a trusted authority to validate the operation is superseded by the use of the consensus protocol. By needing all nodes performing the computation to agree to the output, the ability of bad actors and malicious nodes to tamper with the operation of the contract is entirely negated. However, due to the nature of the contract being executed across the entire network, the contract will not be executed unless the blockchain is sure that the execution will be successful. This prevents wasted operations on the network. The consensus-based nature of this security model means that an attack to the system requires more than 50% of the network to inject any malicious transactions into the blockchain, making it a highly secure decentralized autonomous marketplace (when the continuous expansion and the current size of the Ethereum network are considered).

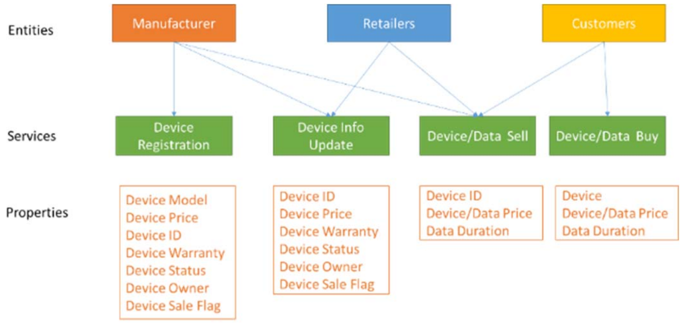

Figure 4. Smart contract logic.

Figure 5 shows an example of how cloud storage can be used. We next describe current implementation of the services in our platform.

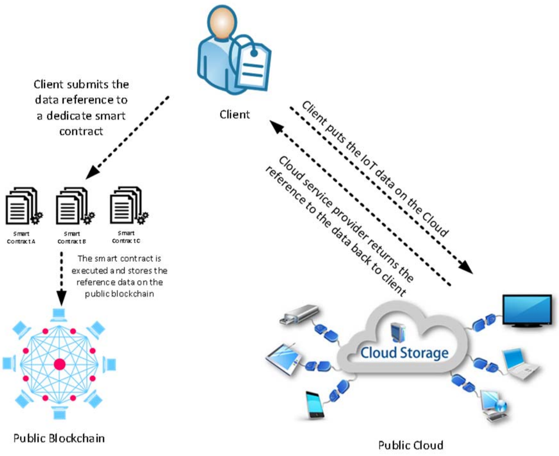
Client
Client submits the data reference to a dedicate smart contract
Smart Contract A
Smart Contract B
Smart Contract C
The smart contract is executed and stores the reference data on the public blockchain
Public Blockchain
Client puts the IoT data on the Cloud
Cloud service provider returns the reference to the data back to client
Cloud Storage
Public Cloud

Figure 5. Our current implementation.

Manufacturers are the entity that produce new IoT devices. They have three functions: Registration, Update, and Sale. The registration service is responsible for registering a newly made device onto the blockchain. Actually, it creates device information containing the metadata related to the device including device model, device ID, and device warranty information. Manufacturers also need to update the device state, which indicates the device can be traded on the market. This update service allows manufacturers to update the device information including the device owner, device price, device warranty, and device status. For instance, when a manufacturer finds a specific product has a safety defect and wants to recall the product, it can set the device status to untradeable. Customers need to return them to the manufacturer; otherwise, they lose the rights to trade them in the platform. The sale service allows the manufacturer to put a specific device on the market for sale. A retailer can buy the device from manufacturers. This service can only operate when the manufacturer sets the device for sale indicator as true.

Retailers have the following three functions: Buy, Update and Sell. The buy service enables retailers to buy a specific device from manufacturers with an expected price. Once the agreement is made between a retailer and a manufacturer, the manufacturer transfers the product's ownership to the retailer. The update service allows the retailer to update the device information including the device owner and the device price. This service allows the retailer to set the price for a given device and allows the retailer to transfer the ownership of the device to another retailer or customer when the ownership transferring agreement is achieved. The sell service allows the retailer to put the specific device on the market, enabling customers to buy the device from the retailer. The retailer needs to set a reasonable price for the device before putting it on the market.

Customers also have three functions: Buy, Update and Sell. The buy service enables a customer to buy a specific device or data from a retailer with the expected price. Once the agreement is achieved between the customer and retailer, the retailer transfers the device's or data's ownership to the customer. The update service allows the customer to update the device/data information including the device/data owner, device/data price. The device/data owner can also set the price for a given device/data and allow the device/data owner to transfer the ownership of the device to another customer when the ownership transferring agreement is achieved. The sell service allows the customer to put a specific device/data on the market; thus, customers can buy the device/data from another customer.

FUTURE RESEARCH CHALLENGES

Through the case study, we demonstrated how a blockchain can be applied to develop a trusted device and data trading platform for the IoT ecosystem where different entities can cooperate with each other. However, blockchain is not a panacea to resolve the trust issue in IoT environments. There are some challenges that still need to be studied. We outline some research challenges and potential future works below:

Data Privacy. Currently there is a dilemma/trade-off between public verifiability and privacy. In a trustworthy data trading platform, the trust is established by providing verifiability. The challenge is how to protect privacy of data, device and individuals without losing the verifiability property. One simple solution is to encrypt all data on the blockchain. This might help to address the privacy problem, but the data cannot be verified by other validation nodes. One potential avenue to further this research is use of Zero-Knowledge-Proof (ZKP). However, the computation overhead is often cited as a key problem in using ZKP.

Delegating trust. Another challenge of any effective trust model is trust delegation. In practice, it means how one can practically delegate trust to someone else. For example, Bob brings a device home and he claims/registers it as his device, perhaps with a straightforward method. Bob is the sole person who can control it and is privy to the data it collects. In certain circumstances, Bob may want to give others access to his device. There needs to be a scheme to ensure that operation can be done reliably and Bob has a full understanding of the implications. The challenge is how to support a trust delegation function without violating underlying security and privacy.

Dynamic Access Control. The access control mechanism is widely used to control access to the data. These methods have been directly applied to IoT environments. However, the IoT system is very dynamic and it operates in a context. For example, an emergency doctor might be able to access the IoT health data or the IoT health data becomes accessible from the IoT devices worn by patients when they are in the emergency room. In essence, it is difficult to predefine all potential access control rules. The device itself should have a mechanism to generate access control rules dynamically based on the contextual information. For example, a drunken driver would not be able to start the car. Though there has been some progress in this area, further research is needed to build a reliable dynamic access control mechanism for IoT applications.

IoT device identification. For the data and device trading platform to function properly, IoT devices that are trusted need to be identifiable. This requires an easy to use identity management system to be made available for all IoT devices at all times. However, the identity management systems (such as username/password pairs, and X.509) are invented for general purpose identification and are thus inadequate and rarely address many known issues that exist in the IoT environment. There is no defined and accepted standard for device identification management in the IoT environment. One of the key features of such a device identification mechanism should be automatic discovery of devices. The challenge is as soon as the device is powered and in operation, it would be discoverable (without violating the underlying security and privacy).

Human-centric trust model. Human-centric trust models are another research topic, which means a trust model aimed at giving effective administration of security and privacy not to computing professionals, but to average users. This is a cross-cutting topic to all of the challenges stated above. For example, a human-centric trust model can be designed for people to sensibly delegate the controls of data and device to others with full understanding of security and privacy implications. The goal of a human-centric trust model is to let the service itself evaluate the security risks and apply the security policies according to the potential attack; thus, an average person can enjoy the same device security levels as security professionals.

Developing a holistic benchmarking kernel. Understanding performance bottlenecks of blockchain-based large scale IoT application systems remains a challenge, hence it is useful to identify benchmark kernels that are relevant for testing particular aspects (e.g., overlay networking, consensus protocol, querying) of the blockchain. Existing benchmarking literature in the context of IoT systems is limited as they are largely focused on studying the scalability of data processing programming models. For instance, the benchmark (kernels) that are available in context IoT systems focus on following data processing programming model aspects (not applicable to benchmarking performance of blockchain):

- **Edge layer.** TPCx-IoT (for data aggregation, real-time analytics & persistent storage), Google ROADEF & Linear Road benchmarks (for stream processing).
- **Cloud layer.** TeraGen, TeraSort, TeraValidate, and BigDataBench (for batch-oriented processing).

Hence, creating benchmark kernels that can test different aspects of the blockchain, and more importantly, identify performance bottlenecks and dependencies need more attention from the research community.

Scalable Search and Communication. Developing a scalable protocol for searching data blocks and smart contracts within a large scale blockchain network remains a challenge. Existing search and consensus communication protocol adopted by the state of the art blockchain networks are based on a complete broadcast routing algorithm. Drawbacks for broadcast-based routing include high network communication overhead and non-scalability. Hence, the future investigation will need to develop scalable methods for search and consensus communication protocol. To this end, one possible research direction can be to interconnect peer nodes in the blockchain network based on Distributed Hash Tables (DHTs) overlay. In contrast, to broadcast based peer-to-peer systems, DHT overlays (e.g., Chord, Pastry, and Tapestry) have been proven to be more scalable as the size of the network grows.

Solutions to these research challenges will form a building blocks and contribute to components required for building a trustworthy data and device trading platform. We believe the blockchain technology is key to finding appropriate solutions to these challenges as it provides a basic foundation for trust in an untrusted, distributed IoT environment.

CONCLUSION

Many our lives are influenced by IoT-driven data science applications, ranging from precision health/medicine to self-driving cars. While enjoying the convenience and benefits brought by IoT devices, we must also face the challenge of trusting such IoT ecosystems. Traditionally, a trusted party performs a supervisory role in data collection and analysis. Blockchain, which is designed to remove the trusted third-party in a decentralized system, is an ideal solution to resolve the trust issue in IoT ecosystems. With the help of blockchain, different parties can trust and verify the data. Additionally, the ownership of IoT devices and their related data can also be traced. Though the blockchain can resolve the trust issue, it is not the panacea to every IoT challenge. A number of research challenges need to be addressed including data security and privacy on the blockchain, trust delegation, device identification, discovery, and authentication.

REFERENCES

1. R. Ranjan et al., “The Next Grand Challenges: Integrating the Internet of Things and Data Science,” *IEEE Cloud Computing*, vol. 5, no. 3, 2018, pp. 12–26.

2. Z. Whittaker, "US cell carriers are selling access to your real-time phone location data," *ZDNet*, 14 May 2018; www.zdnet.com/article/us-cell-carriers-selling-access-to-real-time-location-data.
3. J. Voas, *Network of 'Things'*, NIST Special Publication 800-183, NIST, 2018; https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-183.pdf.
4. B. Celler et al., "Impact of At-Home Telemonitoring on Health Services Expenditure and Hospital Admissions in Patients With Chronic Conditions: Before and After Control Intervention," *Journal of Medical Internet Research*, vol. 5, no. 3, 2017, p. e29.
5. A. Gervais et al., "On the security and performance of proof of work blockchains," *Proc. ACM SIGSAC Conference on Computer and Communications Security* (CCS 16), 2016, pp. 3–16.
6. S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008; https://bitcoin.org/en/bitcoin-paper.
7. D. Hopwood et al., *Zcash protocol specification*, technical report Tech. rep. 2016-1.10, Zerocoin Electric Coin Company, 2016.
8. M. Castro and B. Liskov, "Practical Byzantine fault tolerance," *Proc. 3rd Sym. Operating Systems Design and Implementation* (OSDI 99), 1999, pp. 173–186.
9. C. Cachin, "Architecture of the Hyperledger blockchain fabric," *Workshop on Distributed Cryptocurrencies and Consensus Ledgers*, 2016.
10. S. King and S. Nadal, "Ppcoin: Peer-to-peer crypto-currency with proof-of-stake," August 2012.
11. A. Kiayias et al., "Ouroboros: A provably secure proof-of-stake blockchain protocol," *Advances in Cryptology – CRYPTO 2017*, Katz J., Shacham H., Lecture Notes in Computer Science, vol. 10401, Springer, 2017.
12. *NEO white paper*, white paper, Neo, 18 May 2018; http://docs.neo.org/en-us/.
13. *REDDCoin*; https://reddcoin.com/.
14. I. Eyal et al., "Bitcoin-NG: A Scalable Blockchain Protocol," *Proc. 13th Usenix Conf. Networked Systems Design and Implementation* (NSDI 16), 2016, pp. 45–59.
15. Y. Sompolinsky and A. Zohar, "Secure high-rate transaction processing in bitcoin," *Financial Cryptography and Data Security*, Böhme R., Okamoto T., Lecture Notes in Computer Science, vol. 8975, Springer, 2015.
16. Y. Sompolinsky, Y. Lewenberg, and A. Zohar, "SPECTRE: A Fast and Scalable Cryptocurrency Protocol," *IACR Cryptology ePrint Archive*, 2016, p. 1159.
17. I. Bentov et al., "Tortoise and Hares Consensus: the Meshcash Framework for Incentive-Compatible, Scalable Cryptocurrencies," *IACR Cryptology ePrint Archive*, 2017, p. 300.
18. Y. Gilad et al., "Algorand: Scaling byzantine agreements for cryptocurrencies," *Proceedings of the 26th Symposium on Operating Systems Principles*, 2017, pp. 51–68.
19. V. Buterin, *A next-generation smart contract and decentralized application platform*, white paper, 2014.
20. *LiteCoin*, 2018; https://litecoin.com/.
21. *Ripple*, 2018; https://www.ripple.com/.

ABOUT THE AUTHORS

Bin Yu is a PhD student at Monash University/Data61 CSIRO. Contact her at bin.yu@monash.edu.

Jarod Wright is an industrial trainee at Data61 CSIRO. Contact him at Jarod.Wright@data61.csiro.au.

Surya Nepal is a principal research scientist at Data61 CSIRO. Contact him at Surya.Nepal@data61.csiro.au.

Liming Zhu is a research director at Data61 CSIRO. Contact him at Liming.Zhu@data61.csiro.au.

Joseph Liu is a senior lecturer at Monash University. Contact him at Joseph.Liu@monash.edu.

Rajiv Ranjan is a chair professor in Computing Science and Internet of Things at Newcastle University. Contact him at raj.ranjan@ncl.ac.uk.