2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)

Dependable Public Ledger for Policy Compliance, a Blockchain Based Approach

Zhou Wu
MSCS
Marquette University
Milwaukee, WI, USA
zhou.1.wu@marquette.edu

Andrew B. Williams
HEIRL, School of Engineering
University of Kansas
Lawrence, KS, USA
andrew.williams@ku.edu

Debbie Perouli
MSCS
Marquette University
Milwaukee, WI
despoina.perouli@marquette.edu

***Abstract*—The ever increasing amount of personal data accumulated by companies offering innovative services through the cloud, Internet of Things devices and, more recently, social robots has started to alert consumers and legislative authorities. In the advent of the first modern laws trying to protect user privacy, such as the European Union General Data Protection Regulation, it is still unclear what are the tools and techniques that the industry should employ to comply with regulations in a transparent and cost effective manner. We propose an architecture for a public blockchain based ledger that can provide strong evidence of policy compliance. To address scalability concerns, we define a new type of off-chain channel that is based on general state channels and offers verification for information external to the blockchain. We also create a model of the business relationships in a smart home setup that includes a social robot and suggest a sticky policy mechanism to monitor cross-boundary policy compliance.**

***Index Terms*—Blockchain, state channels, policy compliance, sticky policy, GDPR, data verification, trust, social robot, IoT**

I. Introduction

Over the last few years, both the general public and legislators have become more aware of the wide extent to which personal data is being collected and used by novel cloud services and smart devices. Major reported breaches, such as the Facebook-Cambridge Analytica and the Equifax incidents, caught the public's attention and led some users to reconsider their online behavior (e.g. delete a social media account). On the legislative side, the European Union put into effect a new law called the General Data Protection Regulation (GDPR) in May 2018. GDPR recognizes eight new rights for data subjects and requires service providers to adhere to stricter data use practices. On January 1, 2020, the California Consumer Privacy Act of 2018 (the "Act") will take effect. The Act defines four consumer rights, regulates disclosures made to consumers and has similarities with GDPR, although it is more limited.

Responding to the amounted pressure for better consumer privacy protections, companies are expected to spend significant resources in order to comply with the new regulations and avoid fines. However, at the technology level, it is not clear what kind of tools and techniques will provide the industry with solutions that are transparent, yet efficient and cost-effective. EU member countries have recently started enforcing GDPR fines, while companies that have not yet had enough resources to update policies and build new tools may choose to move out of a market or discontinue popular products that are hard to upgrade according to the new regulations.

During the past year, most major consumer facing companies that operate with personal data, like Google, Facebook and Microsoft, have issued notifications regarding the changes to their privacy policy. Although GDPR only applies to EU citizens, major companies have announced that they will hold the same privacy standard for all of their users regardless of location. As before GDPR took effect, the consumer often faces the dilemma of consenting to their use of private data, with varying amounts of control over the data shared, or deny a service altogether. Well established companies are better positioned to persuade the user of the importance of the service over the sharing of private data. On the other hand, startups or companies entering a new market may face stronger resistance from the consumers.

Our vision regarding the technical solutions that the industry should adopt to comply with privacy requirements, regulation or consumer driven, is guided by the following remarks:

- we argue that some industry players will benefit from a public record of policy compliance that allows them to build reputation and gain consumer trust;
- we argue that a data provenance view that (a) is constructed through the collective traces of multiple stakeholders, and (b) is undisputed by all involved parties is critical in proving compliance in cases where the data travels across administrative boundaries.

In this paper, we describe the architecture of a blockchain based system that records activity digests of organizations operating on private data. We assume that each party has defined the privacy policy rules that should govern their organization's access to consumer data and is able to determine whether a specific action is a violation. We identify transaction verification as a critical problem when using the blockchain as the public ledger for non-cryptocurrency applications and design a new type of off-chain channel that allows two parties to approve or challenge specific data access before

This material is based upon work partially supported by the National Science Foundation under Grant No. 1657548.

2575-8411/19/$31.00 ©2019 IEEE
DOI 10.1109/ICDCS.2019.00187

submitting a message digest to a public blockchain. The off-chain channel is employed to address the scalability limitations of blockchains.

To motivate our design and support the second underlying premise of our vision, we present a model of the relationships that exist among the various physical and business entities involved in a home setting with a social robot and Internet of Things (IoT) devices. During the last decade, we have observed a rapid increase on the number of companies announcing the development of a product promoted as a social robot (e.g., Jibo, ElliQ, Olly, BIG-i, Zenbo). Such a robot is supposed to act as a personal assistant with a "personality" that makes it feel like a human companion [1]. A social robot is designed to reside in homes and has unprecedented potential to affect people's privacy.

Therefore, a transparent and auditable public ledger recording an organization's performance on data protection seems desirable to the industry sectors in which information technology is a significant building block. The contributions of this paper are towards a two-fold vision: (a) achieving verifiability even when combining a blockchain with off-chain transactions, and (b) providing a mechanism for small and medium sized organizations to reduce the costs required for regulation compliance and increase the trust of potential clients. Although the proposed blockchain based mechanism is inspired by the challenges faced by industry and non-profit organizations in the area of policy compliance, it is more generally applicable. This mechanism could be employed to increase the scalability of any blockchain based system, such as cryptocurrencies and smart contracts.

The rest of the paper is organized as follows. Section II describes influential related work. Section III presents the complicated business relationships in a smart home environment with IoT devices and a social robot acting as the central controller. Section IV discusses the privacy issues raised in such a smart home environment. Section V presents the architecture of a dependable public ledger for policy compliance, which is our main contribution. Finally, Section VI offers concluding remarks and ideas for future work.

II. Background and related work

A. Verification in a Blockchain Public Ledger

The blockchain (BC) is a promising technique to implement a public ledger in that each user can view and verify the transactions published on a BC network. Any transaction is confirmed and accepted by all nodes in the network via a mining process. During this process, the transaction is added into a block that is finalized by a proof of work. Once a transaction is confirmed in this manner, there is high probability it is immutable. This desirable property stems from the fact that any new block is generated from the hash of a previous block. Thus, changing a historical block would need to be accompanied by regenerating all following blocks as well [2]. A successful attacker would have to control a significant portion of the computational power of the network. An attacker with 50% of the mining power will succeed in forking the BC with positive probability [3]. Several copies of the entire BC are kept by some nodes in the network, thus the BC inherently introduces data redundancy.

The success of BC in cryptocurrency [2], [4] relies not only on its technical design, but also on its incentivizing strategy for encouraging correct behaviors. In Bitcoin, the most successful BC application, honest miners receive financial rewards. Misbehaviors, such as forking the BC or selfish mining, increase the risk of mined blocks being discarded, which means decreased revenues. Misbehaviors are also discouraged by the fact that if the market is dominated by such misbehaviors, the market confidence will be impacted, and thus the value of the currency will be diminished [3].

Considering the Bitcoin application, the network verifies transactions to determine whether an address owns enough coins to complete a transfer. What will be included in the blocks are the transfer of coins or the generation of new coins. In other words, the Bitcoin "space" is closed. Coins cannot be injected into the network from the external world. The only source of coins is the mining process. There are no transactions that transfer coins which are not derived from mining. The "coin", or its equivalent, is the only verifiable object in a BC. The BC does not provide any tools for verification of external information. For example, if we publish weather records on the Bitcoin network, the nodes cannot verify the weather information we provide through the transaction. What the nodes do is to verify if there is enough coins to pay the transaction fee and accept any other information carried by the transaction submitted.

Ethereum [4] allows the BC network to execute arbitrary user scripts as smart contracts with a fee, which enhances the ability of verification. However, the validity of external information still depends on the reliability of the information source. In a smart contract implementation, Hawk [5], an additional private layer is introduced to verify the external proof of contract compliance, where the BC only takes money transfers as transactions. Therefore, though an external message can be included in BC and is immutable once finalized, such a message is not necessarily reliable, since it can be an arbitrary string and the BC network cannot verify it.

Another problem related to the verification of the transactions is the immutability of historical records. The immutability is a desirable property for a ledger in that the finalized records are supposed to be reliable. However, it could be problematic regarding the technical environment specified by the BC network. Consider a scenario in the weather recording case: an incorrect record is published to the ledger. Because the BC network is not able to verity its correctness, the wrong information would be accepted other than detected. In this case, the immutability of the BC will obstruct efficient corrective actions: records can not be modified in an append-only ledger. One way to correct wrong information is to announce a replacement of the incorrect record. This method is flawed in two aspects: 1) it wastes the expensive storage resource in the BC system (i.g both the incorrect information and its correction will be permanently kept in the BC and

duplicated among the mining nodes); 2) It is difficult to reconstruct the correct records.

Finally, the volume of data records poses scalability issues for a BC. Placing all records into blocks generates very frequent transactions that could disfunction a public BC network. For instance, it takes 10 minutes on average for a Bitcoin block to be confirmed the transaction. Due to the randomness of the mining process, it usually takes longer to finish the transaction. For a Bitcoin transaction to be confirmed, it is expected to take an hour, and the overall throughput of the network is limited to around 7 transactions per second [6]. Existing solutions for scaling blockchains consider structures with off-chain databases and on-chain hashes, in which the reliability of the off-chain databases remains questionable.

B. Off-chain Payment and State Channel

Off-chain payment channels aim to improve the scalability of BC cryptocurrencies for fast and frequent payment processing [7], [8]. In general, two parties deposit coins into a shared multi-signature address to open an off-chain payment channel. After opening, the two parties can make payments to each other through an agreement on the distribution of the deposit coins without generating any on-chain transactions. The agreement is in the form of a commitment transaction. For example, if Alice initiates a payment to Bob via an off-chain payment channel, she will sign a transaction indicating the final balance. This transaction is a commitment transaction which is not broadcasted to the network immediately. New payment will replace a preceded commitment transaction. At closure, the BC network will take the latest commitment transaction and redistribute the deposit coins.

Unless one party could successfully forges the signature of its counterpart, a dishonest party is only able to cheat by posting an expired commitment transaction to close the payment channel. Therefore, any transaction for channel closure should be finalized after a timeout enough for the counterpart to react and potentially dispute. A proved cheater is punished economically in a way defined by the contract.

A disadvantage of this original form of payment channels is that the payment channel is pairwise. Only the participants of the transaction that setup the payment channel could pay each other through it. Thus, a main stream of research on this area aims to connect existing payment channels into an off-chain payment network. If Alice wants to pay Bob, they do not have to setup a payment channel between them, as long as there is a user (we may call her Chris) having payment channels with both Alice and Bob. Alice pays Chris, then Chris pays Bob. Any number of nodes can be added to the chained payment, thus any pair of users in this network could pay each other with establish additional channels.

Another direction of payment channel research is to generalize it into a so-called state channel [9]. The participants of a state channel monitor and operate with some states in concern. In payment channels (which are specific cases of state channels), the interesting state is the deposit paid by the participants. The generalized state channel could accept any variables as the states. This scheme has the potential to broaden the adoption of BC application in areas other than cryptocurrencies, but this potential has not yet been adequately explored. *Raiden* is the most prominent project that implement state channels, but currently it focuses on the implementation of payment channels via this generalized framework. Besides the scalability improvement, the off-chain payment channel, or state channel, guarantees the reliability of external information by posing probable economic loss, if caught behaving dishonestly.

C. Sticky Policy

Policies can stick to data to define allowed usage and obligations as it travels across multiple parties, platforms, or administrative domains, enabling users to improve control over their personal information [10]. *Sticky policy* is a potential approach for accountable and enforceable policies [11], [12]. A similar idea has been used for tracking data flow within cloud infrastructures [13], [14]. A *sticky policy* adopts a data-centered approach that encloses allowed methods with the data object. This scheme could be further extended to define any allowed operation in the descriptor. Sundareswaran et al. proposes a logging system for data sharing following this paradigm [15]. In their work, the users' data is encapsulated with executable code in JAR files.

There is a tradeoff between storage overhead and universal applicability of the *sticky policy*. For instance, consider a system for information flow control that uses a tagging mechanism to identify the policies applied to specific user data within a PaaS cloud [16], [17]. Since the policy recognition and enforcement system is embedded within the cloud infrastructure, the tag associated with the user data is light weight. However, this policy enforcement cannot be applied when data has to travel across the boundary of cloud infrastructures (e.g from EC2 to Azure). If the enforcement code is attached to the data [15], [18], the policy application can be ensured as the data travels through different cloud infrastructures at an overhead probably larger than the data in concern. Ideally, the most effective way to implement *sticky policy* is by a protocol standard in which a header is defined as the policy descriptor, and the processing methods are defined for the policy agent. We follow this paradigm in this paper and assume there is a standard policy descriptor attached to the user data. The corresponding policy agent is deployed through the entities involved in the service provision.

D. Other Related Work

Zhang et. al. categorizes the requirements for dependable, scalable, and pervasive distributed ledgers with BCs and identifies research challenges to achieve this objective [19]. One of the particular issues besides the scalability problem is the transaction privacy. Because of the transparency of the ledger, it is possible to construct an activity graph for a particular address. zkLedger [20] attempts to solve this problem for a public auditable ledger by hiding plain information via *Pedersen commitment* and *non-interactive zero-knowledge*

proofs. Shae and Tsai proposes an approach to transform blockchain into distributed parallel computing architecture for precision medicine [21]. Though different from the purpose of our scheme, it shares the same extension, i.e, to coordinate on-chain and off-chain computation. The core challenge is to keep the on-chain workload light weight. Their work depends on an off-chain control node that could help the on-chain program call off-chain arbitrary code to execute the main computation. This scheme is applicable when the entire computation is owned by a single organization. When multiple administrative parties are involved, the output from other participants may not be trusted and thus a verification process is necessary. There is discussion on the incentive mechanisms underlying the BC based cryptocurrencies [22], [23]. However, their analysis focuses on the explanation of how the existing schemes work rather than provide a quantified method to design incentive frameworks for different application scenarios.

III. Relationship Model

In this Section, we create a model that captures the complicated relationships between different but interrelated administrative authorities in a smart home setup that includes a social robot. Our purpose is to illustrate the administrative borders that data travels and motivate the need for a provenance data trace compiled from the various views of the involved parties for evaluating privacy policy compliance.

In order to perform a risk analysis, we need to consider the ecosystem of service providers, devices, and policies within which the social robot operates. The behavior of the robot is typically controlled through artificial intelligence algorithms in a company's cloud platform, while the robot should also have access to other clouds such as those used by the user to store data (e.g., pictures, documents). A social robot is expected to act as a smart home central controller so that it will be communicating with IoT devices directly or indirectly. Currently, a privacy policy for a networked social robot does not exist and thus cannot be implemented globally nor audited to protect users in work, home and care environments. Additionally, security vulnerabilities of social robots have already been identified [24].

In the typical case, the social robot is not meant to be used in isolation, but in close communication with the robot manufacturer's cloud service. We understand the purpose of this tied connection to be three-fold. First, the robot can be marketed with more competitive pricing without sacrificing computing power, if the robot's hardware is not expected to run all applications, such as intensive artificial intelligence algorithms, but can "outsource" some of the computation to the cloud. Similarly, the robot's storage capacity does not have to be the limit on how many large files, e.g., video recordings, the user can create without running out of space. Some manufacturers could even offer cloud storage at an additional cost. Second, the manufacturer can collect interesting data about the robot use, if the robot is designed to regularly "call home" and write to the cloud's log files. Third, several of the social robot companies release software development kits (SDKs) for interested parties to create add-on robot skills – a concept similar to the creation of apps for smartphones. The robot's cloud could serve as a central place for downloading skills and storing their data.

In Fig. 1, the robot manufacturer's cloud is depicted as cloud A. Apart from the robot's cloud that was just described, the user could also have the robot exchange data with other clouds. Examples include Dropbox, Google Drive, and Facebook. The robot's company might even allow users to download add-on skills from platforms such as Google Play. In Figure 1, these types of cloud services are depicted as a cluster of L clouds.

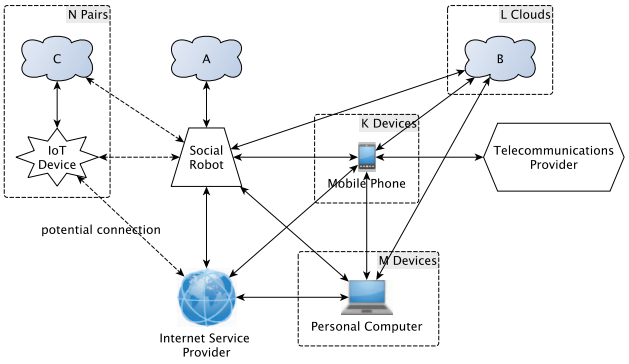

Fig. 1. The relationship model with the social robot in a home setting.

The social robot will be just one of the devices sending or receiving data from the group of those L clouds. Smartphones, tablet and laptop devices found in a household today are often already connected to some of the L clouds. In addition, the user might desire these personal devices to exchange data with the social robot. For instance, the robot could send pictures taken with its camera through the Multimedia Messaging Service (MMS) or electronic mail. Some robots that are marketed not only as a friendly companion but also as a security patrol guard could also push alerts and videos of what they mark as unusual conditions. Similarly, the user might want to share information like the contacts address book from the phone or personal computer to the social robot.

As smart home devices and IoT become even more prevalent, it is reasonable to assume that a home where a social robot is present will likely have a number of IoT devices as well. Examples of currently available smart home devices include smart thermostats, lights, and monitoring systems. Some of the IoT devices may be sending and receiving data to and from a cloud, which is operated by the device's manufacturer or a third party. This cloud collects information from the device's operation and can be used to configure the device dynamically. In Fig. 1, we assume that there are N such IoT device-cloud pairs. Smart home devices often benefit from a central smart home controller. If a social robot exists in a home, it is positioned well to become this central controller. For example, if a smart conductivity sensor reports a non-zero value to the robot, the robot could check the sensor's

surroundings and decide, if there is flooding, to wake up the homeowner.

The introduction of a social robot in a home environment adds a significant degree of complication to the business relationships and network connections enabled in a smart home. Each of the mentioned clouds may be run by one entity as a Software-as-a-Service (SaaS), while a different administrative authority may be supplying the Platform or Infrastructure-as-a-Service (PaaS or IaaS). The same IaaS (e.g., Amazon Web Services) could at times be the layer below two different SaaS clouds (e.g., the social robot's cloud and an IoT device's cloud). Each of the devices, including the social robot, will have their license agreements with potentially varying privacy statements. Tracking the origin and provenance of data, and auditing the information needed to assure that privacy has been preserved can be a challenge with so many authorities involved.

The communication among the devices illustrated in Fig. 1 could involve the cellular network, the Internet Service Provider's (ISP) network or a wireless local network (e.g. WiFi, Bluetooth Low Energy). Currently, many IoT devices do not provide encryption, and the data between such a device and a robot could travel through a network controlled by the Telecommunications Provider or the ISP in clear. Even if the data exchanged is encrypted, we should still consider whether the frequency (or other metrics) of the data traveling from the robot to the network and vice versa could by itself be revealing information.

IV. Nontrivial Privacy Issues

The increased capabilities of a social robot over a traditional IoT device necessitates a privacy policy and the ability to audit this policy across platforms and data transmission channels. These capabilities of the social robot include: move within the environment with precise positioning and orientation in multiple dimensions; listen in on a human conversation; watch human agents through video and computer vision capabilities; act on an environment with physical effectors (e.g. limbs).

A privacy policy describes how a party collects, processes, shares, and manages a person's data and information legally. A person's private information contains anything that can be used to identify a person and can include things such as their name, address, relationship status, finances, health information and buying records. A privacy policy for social robots is required to provide a standard for the design and use of these social, physically mobile and capable, cyber-physical agents because they are inherently personal surveillance agents. A social robot lives in an ecosystem of IoT devices, cloud computing services, apps, and the Internet; a policy and methodology to audit and alert when privacy has been violated are required. The development of this policy will enable manufacturers, users, and computing and internet service providers a standard to manage risk and liabilities of interconnected social robots.

While there is an ISO standard related to personal care robots to ensure the user's safety, currently there are no standards or policy to maintain the user's privacy. The GDPR is not specific to social robots, but has applicable elements. Among its provisions are the right of a person to access the exact information kept by a company regarding that person, and the right to request that all this data be permanently erased from the company's files. The GDPR includes significant penalties for those that are not found in compliance, but the exact long term impact it will have is currently unclear. Nevertheless, the GDPR marks a significant change in the global regulation landscape regarding the privacy of digital assets.

The privacy policy and auditing methodology for social robots should take into account not only data artifacts but also the robot observations, inferences, and machine learning made from single- and crowd-sourced data that can violate one's privacy.

While privacy policies enforce personal data protection in this ecosystem, they usually suffer from lack of transparency and auditability. Policy violation is observed by its consequence, e.g., disclosure of data that is supposed to be confidential. The procedure of the violation has rarely been reported. It is partially because the policy is usually a legal statement rather than a technical protocol that could be examined in detail by certain signals. Suppose a home robot with a webcam records and uploads a video stream in an unexpected way. The incident will be reported only when the house owner observes and recognizes this abnormality. Otherwise, such violation will go undetected.

Policy violations incurred by a social robot consist of both visible actions and invisible computations and communications. For the purpose of transparency and auditability, it is not enough to just take only the result, or the consequence of the violation, into account. The whole process should be under examination. Firstly, some early signals of policy violation are contained in invisible activities. Secondly, transparency and auditability are not only for detecting intended malicious behavior, but also for examining unintended violations, so that improvement could happen, and knowledge could be shared with the community. In the meantime, operation logs that exist for debugging or troubleshooting can provide information for describing violations in detail. This information could serve as the data source for transparency and auditability. However, the log itself includes sensitive private data. The disclosure of the log's contents will put personal privacy at risk.

From another perspective, transparency and auditability also require a permanent unchangeable record that tracks the historical performance of the market participants. On the one hand, it helps the public identify a reliable service provider; on the other hand, honest and well-operated providers could benefit from their efforts on policy compliance.

Auditability is a hard problem in the cloud era, because of the rapid change of structure at the application level and at the underlying architecture level. The presence of a social robot stretches out new challenges, in that robots could perform sophisticated activities whose effect is difficult to measure. The activities of a social robot could be featured as a sequence of individual actions, which, in turn, requires a detailed record

of the whole process of activity used for auditability.

V. Dependable Public Ledger of Policy Compliance

The previous sections have pointed out the research challenges in constructing a BC based distributed system for policy compliance. An average confirmation time of 10 minutes and 7 transactions per second pose scalability limitations on the growth of the BC as a audit log. What is more, the BC cannot verify information external to it.

In this section, we consider a public auditable record of the policy compliance, which can serve as the permanent immutable credit record of the service provider for the customer. A well performing company could benefit from this public ledger by building a trusted reputation. As mentioned previously, the verification of external information is critical for BC based public ledger, especially in the scenario considered in this work, where the information published on the ledger should be reliable. We propose a mechanism called Verifiable Off-Chain Message Channel (VOCMC) that enables the verification of external information and the integration of BC with powerful off-chain computation to overcome the difficulty of scaling the BC.

The VOCMC is derived from the state channel described in Section II-B. When disagreement occurs in an off-chain payment channel, the participants can simply close the channel. However, in the VOCMC we do not adopt this approach, but we preserve the information that leads to disagreements. We introduce a series of negotiations for the parties to resolve the dispute; if these negotiations fail, we still keep a record of the disagreement as a record of policy dispute. The VOCMC is a building block for the public ledger of policy compliance, which combines an off-chain database and an on-chain hash. We assume there exist a set of policies for each party and a mechanism that allows a party to determine whether an action on the data complies with the party's policies. We leverage the *sticky policy* introduced in Section II-C to track the user data and collect evidence.

A. Incentive-Based Trust

There are two kinds of security concerns related to off-chain payment channels. The first is that the deposit balance should be correct; the second is that the publication of the transaction indicating correct current balance state should be guaranteed. The state-of-art work focuses on the publication problem, i.e. how to guarantee the honest participants could publish the valid commitment transactions by the normal closure or by the dispute process, when the counterparts attempt to publish an outdated transaction to rollback the balance state.

On the other hand, the correctness of state has not attracted enough attention, because the correctness of the states seems a natural property in cryptocurrencies. However, the correctness of information is not guaranteed in general applications. In a simple example suppose that Alice pays Bob 10 USD for a sandwich, which can be described in a transaction in three different ways as show in Fig. 2. Although the value of payment is correct in all three cases, the additional information provided by the comment may be correct, ambiguous, or absolutely ignored.

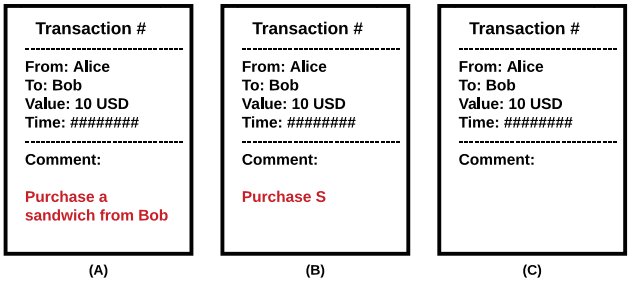

Fig. 2. A: a transaction with comment describing the behavior related to this payment; B: the comment with partial information; C: comment is ignored.

In this scenario, Bob may have no reason to dispute a transaction that contains an ambiguous or empty comment, if the payment amount is correct. In off-chain payment channel, the protocol naturally ensured the correctness of the balance state, because the deposit balance is an incentive-associated information. The participants will automatically take care of the balance when reaching an agreement. However, in applications where the comment field describes information related to policy compliance, the involved parties should have an incentive to check the correctness of information.

Although in theory the security of a BC network requires that none of the participants controls a dominant majority of the mining resources, in practice the formation of mining pools have has challenged this fundamental security assumption. Mining pools are groups of cooperating miners who agree to share block rewards in proportion to their contributed mining hash power [25]. By joining a mining pool, the miners are able to reduce the variance of their mining rewards. At the time of writing, the top three leading mining pools in Bitcoin hold over 51% share of the computing resource; the biggest pool, BTC.com controls 29.6%. Most of the mining pools are concentrated in China with an estimated 81% of the network hash rate [25]. With the presence of mining pools, Eyal and Sirer proposed the selfish mining strategy that allows a pool with one third of the overall hash power to obtain more revenue than its ratio of the total hash power [26].

Nevertheless, such attacks have not been observed. The pools have been benign and followed the protocol so far [26]. The assumption is that the majority miners may avoid strategies that earn more bitcoins but decrease the expected value of their future mining rewards, since a substantial share requires a significant investment to maintain [3].

In cryptocurrency, the participants' incentive is the reward of coins. In off-chain payment channels, the issue of penalties for misbehavior can be used as incentive instead of the reward process. Suppose that Bob pays Alice the correct amount, but files a transaction with a smaller value than expected. Alice will double check the transaction, then close the channel once she discovers this fraud. Bob may lose part of his deposit

according to the contract.

There is an additional factor contributing to the correctness of the states: the direct interest conflict between Alice and Bob. If the state is incorrect, one of the parties is likely to have the incentive to dispute the claim. Therefore, to protect their benefit from pillage, Alice and Bob have a strong motivation to carefully monitor the state. In our applications, the users and the service providers will typically hold opposite interests on data usage. Therefore, an incentive-based mechanism for policy compliance could be applied.

B. Verifiable Off-Chain Message Channel (VOCMC)

The VOCMC is established by n parties who are concerned about the off-chain data that will be used for on-chain execution. We assume the n parties are users of a BC system where a type of cryptocurrency is defined and that each participant holds enough balance of the cryptocurrency in private addresses. Upon the establishment, the n participants have to transfer some units of the cryptocurrency from their private addresses into a n-of-n multi-signature address as the deposits. An n-of-n multi-signature address requires n signatures to authorize a transfer from this address.

The channel is parameterized by (the pseudonyms of) the n parties, $p_{i,i\in n} \in P$, and their deposits, $d_{i,i\in n} \in D$. At any time t, the deposit balance of a participant p_i is $d_{i,t}$. Denote any input to the VOCMC at time t before the expected finalization time T_E as $m_{t,0\leq t\leq T_E}$, the corresponding output to the BC network is $\Phi(m_0, m_1, \cdots, m_t)$ that is generated from all the input till t under method $\Phi(\cdot)$. For convenience, we denote the output at time t as Φ_t. If there is no disagreement on the current output, the parties sign a temporary on-chain transaction $\texttt{TemTran}(t, \Phi_t)$ and each party p_i holds a copy $\texttt{TemTran}(t, \Phi_t, i)$. The message channel is either closed automatically at T_E with the output Φ_{T_E} for on-chain execution, or closed by any party p_i submitting its copy of the latest temporary transaction $\texttt{TemTran}(t_c, \Phi_{t_c}, i)$. The on-chain transaction can only be finalized after a timeout δT enough for counterparts to potentially dispute.

There is an incentive function $\texttt{Inc}(t, \Phi_t, d_{i,t-1})$ that redistributes the deposits according to the current output Φ_t and actions related to it. When dispute occurs (e.g. p_i cheats by submitting an obsolete temporary transaction), the latest valid $\texttt{TemTran}(t_c, \Phi_{t_c}, i)$ will be accepted, and the deposit will be redistributed according to the predefined punishment policy, $d_{i,t} \Leftarrow \texttt{Inc}(t, \Phi_t, d_{i,t-1})$. If the channel is closed regularly, the deposit will be refunded or redistributed according to predefined agreements, $d_{i,t} \Leftarrow \texttt{Inc}(T_E, \Phi_{T_E}, d_{i,T_E-1})$. The dispute and the incentive functions are implemented by a smart contract carried by the temporary transactions.

Different from the off-chain payment channel, there is a particular difficulty when applying state channel for data verification: a participant may intentionally refuse to sign a transaction with messages that may impact its benefit. To do so, the participant could refuse to sign and submit the latest valid temporary transaction to close the channel. Thereafter, the messages will never be published to the ledger.

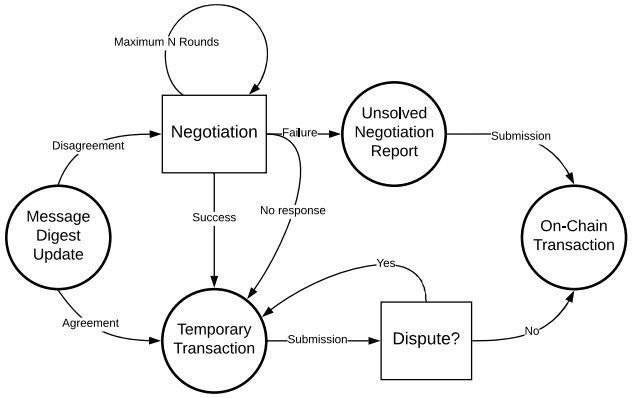

Fig. 3. State transition in VOCMC. Negotiation reports indicate that the information might be unreliable, but the details could help the public make decisions by their own judgement.

We introduce a negotiation procedure to address this issue as shown in Fig. 3. Once a participant refuses to sign a temporary transaction $\texttt{TemTran}(t, \Phi_t)$, the counterpart could initiate a negotiation, $\texttt{Neg}(t, \Phi_t, p_i, t+\delta)$, where $t+\delta$ is the deadline for the participant p_i responding to this negotiation. If p_i does not respond to the negotiation request, $\texttt{TemTran}(t, \Phi_t)$ will be automatically signed. If the negotiation request receives response, but an agreement is not reached, the negotiation stays open with a new deadline for further negotiation. A VOCMC with open negotiations cannot be closed. If the participants do not reach an agreement after N rounds of negotiation (the maximum of rounds allowed), a transaction with information of the negotiation proceedings $\texttt{NegTran}(t + N\delta,$ $\texttt{Neg}(t, \Phi_t, p_i, t+\delta))$ will be automatically signed by every party and submitted to the public ledger. This transaction serves as a report of unsuccessful negotiations. The transaction of negotiation report does not reallocate deposits. Nevertheless, the negotiation report could be a helpful information source for other cuctomers.

The VOCMC effectively removes the scalability limitation when developing BC based applications, because the mechanism can accept any result of off-chain computation as input. If all the participants are honest, the BC network will only process on-chain transactions recording the opening and closing states of the channel with external information verified by the relevant parties. In the case of dispute, the reliability of the input for on-chain process is guaranteed. For the punishment to misbehavior, the VOCMC provides the flexibility to adopt any strategy that is proved by future research, or borrowed from other domain knowledge.

C. The Public Ledger of Policy Compliance

A data provenance view is required due to the cross-boundary problem introduced by the ever frequent storage and process outsourcing. Fig. 4 illustrates such an example. The application employs a social robot that receives oral instructions from a human and takes actions to complete tasks

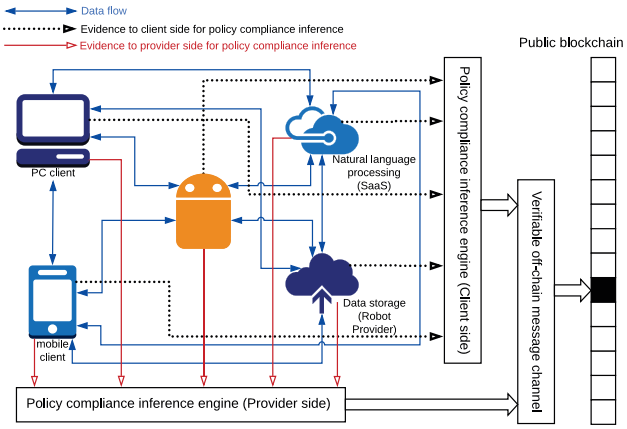

Fig. 4. High level description of the proposed scheme with example

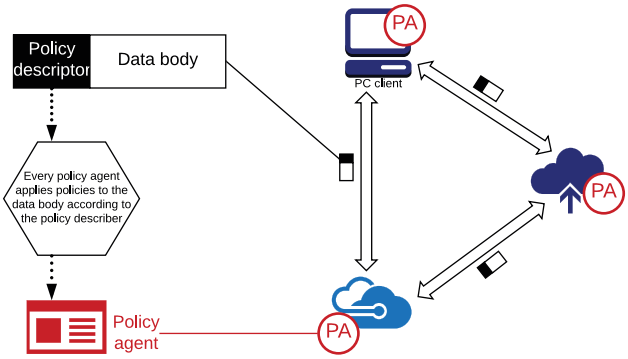

Fig. 5. The sticky policy mechanism that enables policy application and evidence collection in Fig. 4

such as searching for music or traffic conditions. The robot is connected to a PC client and a mobile client for local configuration and control. The robot provider maintains a data warehouse for storage of the user data, and outsources the speech recognition to a SaaS provider specialized in natural language processing. The training data for the speech recognition model is acquired from the data warehouse. When the recognition model is ready, the robot could directly send realtime speech record to the SaaS for analysis.

Our proposed scheme requires the user and the service provider to have already expressed privacy rules and intentions. We assume policy compliance inference engines are deployed at both the user side and the provider side. The scheme should ensure that the required evidence will be sent to both the inference engines. To achieve this goal, we recommend a *sticky policy* based framework to track the data footprint and enforce the policy application. Our *sticky policy* mechanism can be illustrated by Fig. 5 where part of the data flow from Fig. 4 is sketched. The user data will be accompanied by a policy descriptor during its entire life cycle including transmission, storage, duplication, processing, and deletion. The policy descriptor defines the applied policies, or allowed treatments. A policy agent will parse the policy descriptor to determine the policies and configure the execution environment. We do not require that the policy descriptor should carry the execution module, but assume that the mechanism for policy application is deployed within the policy agent at any cloud infrastructure involved in the service provision. In addition, the policy agent is required to have a communication mechanism that will send logging updates to the data owner and the service providers when the data is touched by any program. This log serves as the evidence feeding into the inference engines in Fig. 4.

Because of its inherent flexibility, the *sticky policy* enabled scheme has the potential to be adjusted as policies require. The system also carries the following desirable properties:

- **Consistent cross boundary policy application**. Since any copy of the user data is accompanied by a policy descriptor, it is straight forward to maintain a consistent set of applied policies when the data is moved to another *sticky policy* enabled domain.
- **Global view of data distribution**. A central problem for auditable policy compliance is tracking all the duplication of the user data within the cloud. Data may be copied for various purposes, such as backup, buffering, process outsourcing. Therefore, policy violations might happen not only by malicious behaviors, but also by misconfiguration. The *sticky policy* enabled logging mechanism could help discover all the intentional and unintentional duplications.
- **Fair availability of information for policy compliance inference**. A critical feature of the VOCMC is the timeout δT. Before a hash of the transaction is inserted to the BC, participants are allowed to dispute it. The *sticky policy* mechanism should guarantee independent "push" notification for all the participants. Thus, all participants will be able to examine the transaction and make a decision.

Fig. 6 outlines the high level design of our proposed public ledger of policy compliance. The logging mechanism independently provides evidence to the policy compliance inference engines on both the user side and the provider side. According to the outputs of the inference engines, the user and the provider reach agreement on compliance or violation. Finally, the hash of the agreement appears on the BC.

We recommend that a public and not a private BC system is adopted for the ledger. There are two benefits associated with a public BC for policy compliance. First, the public record allows service providers to build trust with their customer base, especially in cases of startups and companies entering a new market. Second, the major benefit of a BC is its immutability with no trusted third party involved. This property depends on the assumption of the independency of the miners and the fact that no (group of) miner controls a dominant computing power. This assumption is easier to hold for a public BC, but not for private (or permitted) BC. The difficulty in adopting private BC as the ledger is that it essentially introduces a trusted

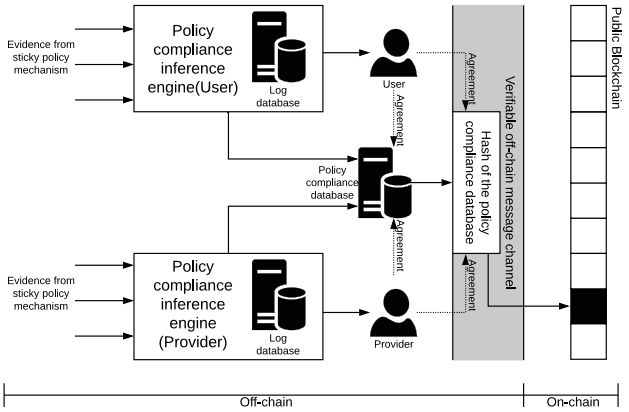

Fig. 6. The structure of the public ledger: off-chain database, on-chain hash

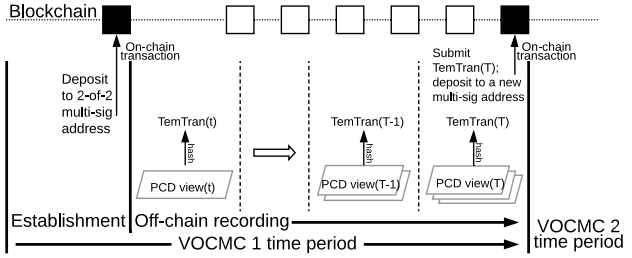

Fig. 7. High level description of the public ledger with example of continuous service

authority. The private chain is owned by an organization and the miners have to get permission from the owner. Ricardo et al. [27] points out that the only feasible solution for BC based database with public auditablity is to utilize a private BC for the recording and a public BC for the checkpoint. We hold a similar point of view, but realize that the private BC is a replaceable component, as long as the checkpoint in public BC is reliable. In a setting where a BC is controlled by a trusted authority, the performance of the BC has to be compared with other distributed database systems.

As illustrated in Fig. 6, there are two levels of off-chain databases: i) the database of evidence collected by the *sticky policy* mechanism, ii) the database of policy compliance, which contains the outputs of the inference engines. The user and the provider reach agreement on both the updates to the policy compliance database and the renewed hash of the database. Note that only the hash is present in the VOCMC, because by definition the agreements obtained in the VOCMC will be in form of temporary transactions that would be published on the public BC. As already mentioned, we follow the structure combining an off-chain database and an on-chain hash, thus only the hash could be included in the on-chain transactions. The policy compliance database can be maintained by the service provider or a third party similar with the TransUnion® structure for personal credit record.

D. VOCMC Setup for Public BC

Suppose the Bitcoin or EthereuBC is the ledger for the on-chain transactions. The user and the provider transfer deposits into a 2-of-2 multi-signature address to open a VOCMC for pairwise recording. The deposit could be part of the payment for the service provider, which would be transferred to the provider's address at the end of the service provision according to the incentive model. The incentive model is implemented by the smart contract. Every time an update to the database arrives, all the parties should sign a temporary transaction that contains the hash of the renewed view of the database, if an agreement is reached on the updated states. Regarding the finalization, the channel is closed when finalizing a record with redistribution of the deposit. The reason for the decision is firstly that the on-chain cost is almost the same no matter what kind of transaction is processed, and secondly that finalization could be caused by dispute where deposit must be redistributed. If the finalization is triggered by the timeout T_E and the service continues, the deposit will be transferred to another 2-of-2 address as opening another VOCMC.

Figure 7 illustrates the ideal case in which all the participants are honest. The VOCMC output to the BC periodically sets checkpoints on the chain. During the finalization, smart contract for the incentive model would be executed by the miners at the cost of the service provider[1].

E. Anonymity

The BC community promotes anonymity as the default for any BC application in order to protect the privacy of the users. Because of the public visibility, the users' detailed activities could be reconstructed, if identities are not properly treated. The basic approach for anonymity in a BC system is to strictly abandon address reuse: an address can only be used for exactly one transaction, regardless of representing a receiver or a sender. Anonymity provides on-chain privacy [5]; transactional privacy is provided on the public BC, unless the contractual parties themselves voluntarily disclose information.

In our scheme, we embrace the anonymity of the data owner, i.e the user of the data service. However, to enable auditability, we intentionally allow the reuse of service provider addresses. A service provider benefits from reusing an address for all activities related to a certain business, since a trackable history record of good performance helps marketing. On the other hand, if a potential customer is given a new address with no publicly available historical record, the customer could suspect misbehavior.

VI. Conclusion and Future Work

We propose an innovative approach for constructing a public ledger of policy compliance. In particularly, we introduce an

[1]This is not required. If the policy compliance database is maintained by a third party, the cost for on-chain computation could be paid by the third party as it joins the VOCMC by a 3-of-3 multi-signature address. Moreover, any operation cost will finally be transferred as part of the service charge to the consumer, thus this is not a essential requirement.

off-chain channel called VOCMC to address the verification of external to the BC information. Apart from the verification of information when there is a combination of on-chain and off-chain system elements, the VOCMC also improves the scalability of the system by limiting on-chain computation. The scenarios we considered are motivated by the privacy policy compliance issues that arise in a home setting including IoT devices and a social robot as a central controller.

A potential attack to our scheme is the *silent agreement*. For example, a provider might make a deal with the customer so that a violation of the policies would not appear on the public ledger. A possible solution to this problem is to introduce into the VOCMC a witness who acts on behalf of the public interest. The challenge in this approach is to avoid the limitations that a centralized authority introduces. The mining nodes of the blockchain network are potential candidates for playing the role of the witness, but to leverage the nodes requires on-chain computation.

While traditional regulation of privacy preservation (i.e. GDPR) focuses on control of customer data, the application of AI and IoT devices adds new dimensions of attacking surface that need an upgraded policy. For example, [28] describes a scenario of a hacked air-conditioner triggering another smart controller to open the windows. It is challenging to define policies for such context-dependent scenarios with pre-determined conditions. In addition, detection of this kind of misbehavior is based on non-deterministic algorithms. As future work, we plan to investigate the possibility of identifying policy semantic structures appropriate for this new type of privacy policies and then design a corresponding policy language and a verification mechanism to integrate with the public ledger.

REFERENCES

[1] C. Breazeal, "Socially Intelligent Robots," *Interactions*, vol. 12, no. 2, pp. 19–22, Mar. 2005. [Online]. Available: http://doi.acm.org/10.1145/1052438.1052455

[2] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.

[3] J. Bonneau, A. Miller, J. Clark, A. Narayanan, J. A. Kroll, and E. W. Felten, "Sok: Research perspectives and challenges for bitcoin and cryptocurrencies," in *2015 IEEE Symposium on Security and Privacy*, May 2015, pp. 104–121.

[4] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger," *Ethereum Project Yellow Paper*, vol. 151, 2014.

[5] A. Kosba, A. Miller, E. Shi, Z. Wen, and C. Papamanthou, "Hawk: The blockchain model of cryptography and privacy-preserving smart contracts," in *2016 IEEE Symposium on Security and Privacy (SP)*, May 2016, pp. 839–858.

[6] K. Croman, C. Decker, I. Eyal, A. E. Gencer, A. Juels, A. Kosba, A. Miller, P. Saxena, E. Shi, E. Gün Sirer, D. Song, and R. Wattenhofer, "On scaling decentralized blockchains," in *Financial Cryptography and Data Security*, J. Clark, S. Meiklejohn, P. Y. Ryan, D. Wallach, M. Brenner, and K. Rohloff, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 106–125.

[7] J. Poon and T. Dryja, "The bitcoin lightning network: Scalable off-chain instant payments."

[8] A. Miller, I. Bentov, R. Kumaresan, and P. McCorry, "Sprites: Payment channels that go faster than lightning," *CoRR*, vol. abs/1702.05812, 2017. [Online]. Available: http://arxiv.org/abs/1702.05812

[9] S. Dziembowski, S. Faust, and K. Hostáková, "General state channel networks," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '18. New York, NY, USA: ACM, 2018, pp. 949–966. [Online]. Available: http://doi.acm.org/10.1145/3243734.3243856

[10] S. Pearson and M. Casassa Mont, "Sticky policies: an approach for privacy management across multiple parties," *IEEE Computer*, vol. 44, no. 9, pp. 60–68, 2011.

[11] M. C. Mont, S. Pearson, and P. Bramhall, "Towards accountable management of identity and privacy: sticky policies and enforceable tracing services," in *14th International Workshop on Database and Expert Systems Applications, 2003. Proceedings.*, Sep. 2003, pp. 377–382.

[12] S. Li, T. Zhang, J. Gao, and Y. Park, "A sticky policy framework for big data security," in *2015 IEEE First International Conference on Big Data Computing Service and Applications*, March 2015, pp. 130–137.

[13] V. Pappas, V. P. Kemerlis, A. Zavou, M. Polychronakis, and A. D. Keromytis, "CloudFence: Data Flow Tracking as a Cloud Service," in *Research in Attacks, Intrusions, and Defenses*, ser. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, Oct. 2013, pp. 411–431.

[14] D. Muthukumaran, D. O'Keeffe, C. Priebe, D. Eyers, B. Shand, and P. Pietzuch, "FlowWatcher: Defending Against Data Disclosure Vulnerabilities in Web Applications," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '15. New York, NY, USA: ACM, 2015, pp. 603–615. [Online]. Available: http://doi.acm.org/10.1145/2810103.2813639

[15] S. Sundareswaran, A. Squicciarini, and D. Lin, "Ensuring distributed accountability for data sharing in the cloud," *IEEE Transactions on Dependable and Secure Computing*, vol. 9, no. 4, pp. 556–568, July 2012.

[16] J. Bacon, D. Eyers, T. F.-M. Pasquier, J. Singh, I. Papagiannis, and P. Pietzuch, "Information flow control for secure cloud computing," *IEEE Transactions on Network and Service Management*, vol. 11, no. 1, pp. 76–89, 2014.

[17] J. Singh, J. Powles, T. Pasquier, and J. Bacon, "Data flow management and compliance in cloud computing," *IEEE Cloud Computing*, vol. 2, no. 4, pp. 24–32, 2015.

[18] J. W. Holford, W. J. Caelli, and A. W. Rhodes, "Using self-defending objects to develop security aware applications in java?" in *Proceedings of the 27th Australasian conference on Computer science-Volume 26*. Australian Computer Society, Inc., 2004, pp. 341–349.

[19] K. Zhang and H. Jacobsen, "Towards dependable, scalable, and pervasive distributed ledgers with blockchains," in *2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)*, vol. 00, Jul 2018, pp. 1337–1346. [Online]. Available: doi.ieeecomputersociety.org/10.1109/ICDCS.2018.00134

[20] N. Narula, W. Vasquez, and M. Virza, "zkledger: Privacy-preserving auditing for distributed ledgers," *auditing*, vol. 17, no. 34, p. 42, 2018.

[21] J. Tsai, "Transform blockchain into distributed parallel computing architecture for precision medicine," in *2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2018, pp. 1290–1299.

[22] J. Chiu and T. V. Koeppl, "Incentive compatibility on the blockchain," 2018.

[23] E. Benos, R. Garratt, and P. Gurrola-Perez, "The economics of distributed ledger technology for securities settlement," 2017.

[24] J. Miller, A. B. Williams, and D. Perouli, "A Case Study on the Cybersecurity of Social Robots," in *Companion of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, ser. HRI '18. New York, NY, USA: ACM, 2018, pp. 195–196. [Online]. Available: http://doi.acm.org/10.1145/3173386.3177078

[25] J. Tuwiner. (2019) Bitcoin mining pools. [Online]. Available: https://www.buybitcoinworldwide.com/mining/pools/

[26] I. Eyal and E. G. Sirer, "Majority is not enough: Bitcoin mining is vulnerable," *Commun. ACM*, vol. 61, no. 7, pp. 95–102, Jun. 2018. [Online]. Available: http://doi.acm.org/10.1145/3212998

[27] R. Neisse, G. Steri, and I. Nai-Fovino, "A blockchain-based approach for data accountability and provenance tracking," in *Proceedings of the 12th International Conference on Availability, Reliability and Security*, ser. ARES '17. New York, NY, USA: ACM, 2017, pp. 14:1–14:10. [Online]. Available: http://doi.acm.org/10.1145/3098954.3098958

[28] T. Yu, V. Sekar, S. Seshan, Y. Agarwal, and C. Xu, "Handling a trillion (unfixable) flaws on a billion devices: Rethinking network security for the internet-of-things," in *Proceedings of the 14th ACM Workshop on Hot Topics in Networks*, ser. HotNets-XIV. New York, NY, USA: ACM, 2015, pp. 5:1–5:7. [Online]. Available: http://doi.acm.org/10.1145/2834050.2834095