Blockchain Enabled Data Marketplace - Design and Challenges

Prabal Banerjee*, Sushmita Ruj†
Cryptography and Security Research Unit
Indian Statistical Institute
Kolkata, India
Email: *mail.prabal@gmail.com, †sush@isical.ac.in

Abstract—Data is of unprecedented importance today. The most valuable companies of today treat data as a commodity, which they trade and earn revenues. To facilitate such trading, data marketplaces have emerged. Present data marketplaces are inadequate as they fail to satisfy all the desirable properties - fairness, efficiency, security, privacy and adherence to regulations. In this article, we propose a blockchain enabled data marketplace solution that fulfills all required properties. We outline the design, show how to design such a system and discuss the challenges in building a complete data marketplace.

Motivation

In this information age, data is the new oil[1]. From the biggest online giants providing free access to platforms to companies offering reduced health care premium to employees[2], data is the latent currency. Even information about the population of a country are becoming vital for governance[3]. To an end user, the trade of her data for services is tacit whereas the companies regularly track and trade user data. A simple example would be a recent deal between WhatsApp and Google giving users option to back up WhatsApp chat data to Google Drive without impacting their storage quota[4]. To users, this would seem like a good deal, but the fine print says that the backed up data is saved unencrypted. So Google would be able to read user data stored on its servers.

Apart from the widespread data and service trading between companies and its users, there are specific data marketplaces which facilitate data trade by matching demand with information sources. Information providers or sellers showcase their data to woo potential buyers. Interested buyers search and select the information they want and acquire it in exchange for money. The marketplace (mediator) gets incentive for facilitating the trade and, in some cases, for hosting the data. But as data grow in value, so will the cases of cheating and leaks. A key aspect which the conventional data marketplaces tend to ignore is ensuring fair trade. The involved parties - sellers, buyers and mediators, being strategic players, would often want to collude and cheat in order to gain more money. Also, if the mediator hosts the complete trade, and buyer and seller have no communication, then the mediator is in a position of power. It can equivocate and forge for monetary gains.

Even if such cases are prevented, there are larger regulatory and privacy concerns related to data sharing. Different countries have different privacy laws and hence trade transactions can be pretty complicated. For example, healthcare related data needs to comply with HIPAA (or FERPA). Also, to avoid misuse, the trade history needs to be monitored over time so as to keep tab over ownership and terms of use.

To tackle the aforementioned problems, we aim to introduce a blockchain layer to the system, which is a distributed tamper-proof append-only ledger. This would enable us to keep historical record of transactions and hence ownership, allow regulatory bodies to (selectively) monitor trade, ensure fair trade with smart contracts and disincentivize collusion using proper incentive mechanism.

Use of blockchain for trading physical goods have been studied[5]. While tracking physical entities as states on the blockchain is a tough problem in itself, it is different than trading data. This is because distribution of data to peers may leak information to parties which is not desirable. Also, keeping enormous amounts of data on the blockchain is not possible because of the severe scalability and usability impact. Control of valuable information on the chain so that only the intended recipient gets access is one of the problems we aim to discuss in this article. We would also like to describe the market design, required properties and how existing technologies like blockchain come together to help create an ideal data marketplace. On top of this, we would also try to identify the major challenges that remain in this space.

Requirements

Digital marketplaces have been studied for a long time, with[6] or without blockchain[7], [8]. Let us enumerate the desirable properties of any data marketplace. Any marketplace primarily consists of two classes of players participating in the trade - the seller $S$ and the buyer $B$. There can be other middlemen who facilitate the trade, such as a marketmaker $M$, but even with their presence, the trade can be seen as a two step trade between $S$ and $M$ followed by $M$ and $B$.

The main features of a data marketplace are:

1) Fairness: Before executing a trade, the commodity and the price needs to be agreed upon. In our case, the commodity being data, we can define a predicate $\Phi$ that needs to be satisfied by the data. So, $\Phi$ and price $P$ are decided by $S$ and $B$ before trade. We call the trade fair, if either $B$ receives the data satisfying $\Phi$ AND $S$ receives $P$, or neither receives
any of it. Any party should be able to pull out of the trade unilaterally, which ensures the timeliness property.

An example of $\Phi$ would be a hash value check so as to ascertain the correctness of the data received. Imagine a file $F$ being traded where both parties agree that for some cryptographic hash function $H$, $H(F) = h$. Then, $\Phi$ is chosen such that $\Phi(F) = 1$ iff $H(F) = h$. Under the assumption that $H$ is collision-resistant, it is hard for a malicious party to find $F'$ such that $H(F') = h$, i.e., the malicious party can cheat the agreement over $\Phi$ with negligible probability.

2) **Transparency, Privacy and Security**: The information that is being traded must not get leaked to any other party except $B$, that too only if the trade was successful. There should be public log as to ascertain the ownership of data and assurance that at least through this system, transferring data or ownership without keeping log should not be possible. This property does not capture piracy as anyone can leak data outside the marketplace which may be beyond the control of the honest actors of the system. We also want transparency in terms of pricing where, in a trade between $M$ and $B$, $B$ should know what was the original price, terms and conditions of the trade between $M$ and $S$.

An example would be Facebook collecting data from users and selling it to third parties. If such a public log of information transfer was available, scandals like Facebook-Cambridge Analytica[1] could have been identified much faster, if not prevented. Moreover, any party buying the information could be held accountable later as she had complete knowledge of the terms of use of personal data between Facebook and its users.

3) **Regulation**: Different types of data have different regulations that may be related to the parties involved in the trade. For example health data have severe regulations like HIPAA. Another example would be a case where an individual had sold her data and wants to know which parties have access to the information currently. The system would need to ensure such regulations are adhered to. Most regulation enforcement can be tricky because of trade across geographical boundaries and conflicting laws between countries/regions. Any breach of information is severe so the marketplace should provide features that inspire trust among the actors.

4) **Efficiency**: For any practical system to have widespread adoption, efficient implementation should be possible. We would call a data marketplace efficient if proposed protocol has comparable speed and requirement compared to state-of-the-art conventional protocols used today.

---

1Facebook gave access to private information of around 50 million users, mostly without consent, to a Professor for academic purposes. The man who sold it to Cambridge-Analytica, which was against Facebook’s policy. Cambridge-Analytica is accused of using the data to affect electoral outcomes across multiple elections in multiple countries. Report: https://www.nytimes.com/2018/03/19/technology/facebook-cambridge-analytica-explained.html
interactions between the players are modelled by the graph. This graph is not stored as a whole by any of the system participants, but can be constructed entirely or partially as and when required from the information on the ledger. Formally, let \( G = (V,E) \) be a directed graph, where \( |V| = n \) is the number of players in the system and for each \( e \in E \), \( e = (u,v) \), \( u \) be the seller and \( v \) be the buyer for that transaction \( e \). We omit the money flow from \( v \) to \( u \) and we assign weights to edges as the price of data. So, \( w(e) = p \) where \( p \) is the price of data being sent from \( u \) to \( v \).

For each \( e_i = (u,v) \in E \), we define \( \Phi_i \) to be the predicate that \( u \) and \( v \) agreed upon, according to property 1. \( w(e_i) = p_i \) be the price of the data \( D \) that is being sold by \( u \) and bought by \( v \). The players, additionally, have access to a global ledger functionality \( \mathbb{L} \), which is the blockchain layer consisting of a smart contract. Figure 2 shows a sample graph.

**Fig. 1.** The four phases of a transaction

**Fig. 2.** A Data Marketplace graph consisting of 100 players. We can think of the saffron vertices as individual data sellers. Blue vertices are data aggregators that collect and compile data from various sources and then sells it. Green vertices can be corporations consuming data.

We outline the process of creation of the graph in Figure 1. Initially, let \( G = (V,E) \) where \( V = E = \phi \). Let \( f() \) be an invertible function. For each interested seller \( u \) and buyer \( v \), the transaction between \( u \) and \( v \) follows the following phases:

- **Phase 1:** \( u \) and \( v \) agree on some predicate \( \Phi_i \) and price \( p_i \).
- **Phase 2:** \( v \) records \( p_i \) on \( \mathbb{L} \).

- **Phase 3:** (a) \( u \) sends \( f(D) \) to \( v \) and \( f^{-1} \) to \( \mathbb{L} \). (b) \( v \) verifies \( \Phi_i(D) = 1 \) and sends confirmation to \( \mathbb{L} \). (without access to \( D \))

- **Phase 4:** (a) \( v \) fetches \( f^{-1} \) from \( \mathbb{L} \). (b) \( p_i \) is sent \( u \) by \( \mathbb{L} \).

We need to note that in Phase 3(b), buyer receives \( f(D) \) and not \( D \). Hence it should verify another statement \( \Phi' \) such that \( \Phi'_i(f(D)) = 1 \Rightarrow \Phi_i(D) = 1 \).

If the above protocol terminates successfully, we modify \( G \) as \( V' = V \cup \{u,v\} \) and \( E' = E \cup \{(u,v)\} \). \( w((u,v)) = p_i \). In case of disputes of over transaction, \( \mathbb{L} \) arbitrates and decides whether \( \Phi_i(D) = 1 \).

As an example, we can think of Alice selling Bob a file \( D \) divided into chunks \([1, \ldots, c]\), for a price of \$5. \( f() \) can be a random permutation over \([1, c]\) and \( \Phi(D) = 1 \) iff all chunks contain a particular bitstring \( s \). Of course, to the buyer, \( f(D) \) is useless but she can verify \( \Phi_i(D) = 1 \) after getting \( f(D) \). Later she is revealed \( f^{-1} \) which she applies over \( f(D) \) to get the desired information. Here, \( u \) is Alice, \( v \) is Bob, edge is of form \((u,v)\) and \( w((u,v)) = 5 \).

To argue that our design can guarantee properties 1-4, we need to give some additional description.

- **Fairness:** For a single buyer-seller trade, under the assumption that either the seller or the buyer is honest, the design can assure strong fairness guarantees. We can use a protocol like FairSwap\[19\] to guarantee fairness. The overall idea of FairSwap is that both \( B \) and \( S \) agree over \( \Phi \) which is expressed as a boolean circuit. Merkle trees are constructed over the boolean circuit and encrypted data, and the root of the Merkle trees are put into the ledger. The encryption key is also stored into the ledger using cryptographic commitments. The scheme is designed in such a way that while decrypting if \( B \) is unable to decipher information as per agreement, it can generate a Proof of Misbehavior and submit to \( \mathbb{L} \) which can then punish \( S \) for wrongdoing. The protocol achieves fairness by running a judge contract on the ledger which resolves disputes and money is held by the judge until the buyer is satisfied or raises a valid complaint (or a time out).
• **Transparency, Security and Privacy:** Every transaction is mediated through $L$ and hence recorded in the ledger. Any party can ascertain data ownership by looking into the blockchain. For security, we use some encryption function in place of $f$ and some cryptographic hash function in $\Phi$. $f^{-1}$ is the key used to encrypt $D$. For ensuring privacy, we need to ensure that $L$ does not have access to $D$ but can still verify if $\Phi(D) = 1$. This can be done using Zero Knowledge Proofs or Proofs of Misbehavior [19]. Privacy is maintained in ZKP as no knowledge about the data is revealed while proving some property. In FairSwap, the data in itself is never sent to the judge contract or any other mediator and hence privacy is preserved. Any other scheme which is fit for our purpose can also be used.

• **Regulation:** Checking whether a data trade follows guidelines can be tricky. Here we assume that regulations can be codified and for each trade (edge $e_i$), let $\rho_i$ be the predicate that decides whether the data exchanged $D_i$ follows the regulations or not. In addition to $\Phi_i$, both buyer and seller inherently agrees to $\rho_i$ when they enter trade. Unlike $\Phi_i$, $\rho_i$ is stored in $L$ as tamper-resistant log. If some $D_i$ does not follow $\rho_i$, then it is on the buyer to report the error and cancel the trade, as done for $\Phi$. If not, the violations can be found out by law enforcement authorities and penalized on retrospect. An example of a simple $\rho$ is a regulation stating that a piece of information cannot be commercially traded. Hence, for such trades, price $p = 0$ needs to be enforced by both parties. If not done, or money is traded off-chain, then the parties can be penalized by law enforcement authorities with the log of $\rho$ acting as an evidence.

• **Efficiency:** The major hurdle for efficiency is the complexity of defining and verifying $\Phi_i$ and $\rho_i$. We could use SNARKs, Merkle trees over boolean circuits or state-of-the-art algorithms for analysis over encrypted data. For simple predicates, all of the above techniques are suitable, but there is still work to be done for complex predicates.

**Example**

Let us consider an example of such a marketplace. A new online medicine portal wants to give targeted ads and regulate area-wise stock and hence needs medicine consumption and health record data. The data sellers might be small medicine shops in every locality. Obviously, there is a gap between the sellers and buyer and hence there will be data aggregators in the middle who will collect the localized data from individual sellers and accumulate the information before selling it finally to the buyer. A sample flow and graph is given in Figure 3.

![Diagram](image)

Fig. 3. Example of a data trade. The flow shows the transfer of information and the graph shows the interactions as per our model.

The last extended example shows that although our design captures most data trades, not necessarily all trade follow our specifications. In particular, we want to have intermediaries in between sellers and buyers. The previous model assumed any data flow to be a chain of two party trades, but now we want to weaken the assumption and introduce intermediaries who do not buy or own data, but just facilitate trade. We aim to extend our model to capture the intermediaries and show it is still secure.

With the changed setup, let us state the roles of players in the trade.

• **Seller ($S$):** $S$ transfers obfuscated information $f(D)$ to a trade mediator and reveals the original information to the buyer in exchange for monetary incentive. She forges information related agreement only with the buyer. The agreement with mediator is to transfer percentage of incentive on successful matchmaking by the mediator.

• **Mediator ($M$):** She collects obfuscated data from $S$ (Potentially collects from multiple sellers and sells the collection to a buyer. But here we consider the simplest case of one seller). She wants to avoid data leak and just wants to earn incentive for successfully facilitating an information transfer.
• **Buyer** *(B)*: B is interested in data D owned by S and wants to acquire the data for some price p, given that it satisfies \( \Phi \). It gets to know about D and S through M and hence willing to give commission to M.

The players interact as follows:

1. S provides M with \( f(D), \rho \) and agree on price \( p_S \) and commission \( c_S \). Agreement is sent to L.
2. B provides M with \( \Phi \) and agrees on price \( p_B \) and commission \( c_B \). Agreement is sent to L.
3. M ensures \( \Phi(D) = 1 \) and sends \( f(D), \rho \) to B for price \( p = \min(p_S, p_B) \). \( \rho, p \) and other parameters of the deal is sent to L.
4. B checks if it agrees to \( \rho \) and proceeds with paying \( p + c_B \times p \) to L.
5. S sends \( f^{-1} \) to B.
6. B sends a complaint with evidence (like Proof of Misbehavior) to L within time \( t \).
7. L checks complaint and penalizes guilty party as per norms of the system. Otherwise, it sends \( (c_S + c_B) \times p \) to M and \( p - (c_S) \times p \) to S.

A pictorial representation of the above is given in Figure 1. To capture such mediated trade relationships in our directed graph, we extend \( G \) to be \( G = (V, E, T) \) where \( T \) is a set of three tuples \( (s, m, b) \) which represent Buyer, Mediator and Seller respectively. Hence, \( s, m, b \in V \). For each tuple, the interaction is as described above. A sample graph is given in Figure 2 for reference.

**Collusion**

As the number of players increase in a particular trade, so does the chance of players colluding with one another. For our original framework, each trade was between two players. As discussed, there exists literature on two party fair trade. But in our extended framework, each trade has three parties. Although, in practice, this can be between multiple sellers selling information to a buyer through a mediator, but here we restrict ourselves to a single seller, buyer and mediator for simplicity. We need to argue that for such interactions, the protocol is resistant to adversarial players and their collusion and hence we iterate over the malicious cases and provide an argument for each situation. We assume here that all players are rational and will act maliciously only for some monetary benefit.

- **Malicious Seller**: A malicious seller \( S^* \) would cheat by sending \( D^* \neq f(D) \) or some \( \rho^* \neq \rho \). Providing \( \rho^* \) makes her vulnerable to retrospective legal actions as a copy of \( \rho \) is kept on the chain. For \( D^* \), it would have to lie again in step 5 and hence in step 6 an honest B would raise a complaint and eventually be penalized.

- **Malicious Mediator**: A malicious mediator \( M^* \) may fail to confirm \( \Phi(D) = 1 \) or tamper \( f(D) \). For the former, B immediately checks and the deal breaks, hence there is no incentive for \( M^* \) to try that. For the latter, B raises complaint in step 6 and S proves innocence using access to \( D \) and agreement in step 1 stored on the blockchain.

- **Malicious Buyer**: A malicious buyer \( B^* \) may attempt to underpay or raise false complaints. If it directly underpays, it will not receive \( f^{-1} \) in step 5, as S checks step 4 before acting in step 5. For false complaints, it should not be able to forge witness. Existing protocols like ZK Proofs and Proofs of Misbehavior already ensure such properties and hence, with high probability, \( B^* \) will be penalized.

- **Colluding Seller and Mediator**: As both the cases of malicious seller and malicious mediator hinges on an honest buyer, hence this collusion brings no new attack vector. The argument for the two cases collectively works for this situation.

- **Colluding Seller and Buyer**: \( S^* \) and \( B^* \) may collude to deny M of his commission. The only way to do so is by trading off-chain as the incentive transfer is handled by the smart contract. While using the system, the collusion cannot deny an honest mediator of her commission, under normal honest majority assumptions of the blockchain platform. Our system is not meant to tackle out of system transactions like piracy and hence we do not address the case where the buyer and seller collude to transact outside the system.

- **Colluding Mediator and Buyer**: \( M^* \) and \( B^* \) might together try to gain access to \( D \) without paying \( S \). But to do that, \( B^* \) either needs step 5 to be executed before step 4 or lie on step 6. \( S \), being honest, will not execute step 5 if step 4 is not complete. False complaint, as argued above, is caught with high probability. Hence, collusion between malicious \( M^* \) and \( B^* \) does not incentivize any of the parties.

**Example with Intermediaries**

In the previous example, we highlighted the need for an extension to the framework to handle practical challenges that arise during data trade. We now wish to provide pseudocodes for the different parties involved in such a minimal trade. For simplicity, we here assume a single seller, buyer and mediator.

The local medicine shop (seller) owns data D which contains fields like \( (id, \text{Age}, \text{Blood Group}, \text{Class of Disease}, \text{Drug Name}, \ldots) \) where \( \text{Class of Disease \in \{Diabetes, Heart Ailments, Psychological Issues\} } \). It wants to sell it at a price \( p_S \) such that no personally identifiable information is transferred.

The buyer B wants medical records related to diabetes, heart and psychological drugs sold in the locality. Hence, it describes \( \Phi(D) = 1 \) iff \( D \) has a field named \( \text{Class of Disease} \) where all entries belong to the set \{Diabetes, Heart Ailments, Psychological Issues\} and dataset \( D \) has at least 10000 rows.

The above example pseudocodes can be made arbitrarily complex to cover multiple scenarios like having \( \Phi = (\Phi_M, \Phi_S) \) where \( \Phi_M \) is only checked by Mediator over encrypted data. \( \Phi_S \) can contain predicates that the unencrypted data should satisfy and \( S \) should terminate trade if \( D \) does not satisfy \( \Phi_S \). Similarly, the ledger functions can be extended to punish either \( M \) or \( S \) or both, depending on Proof of Misbehavior. Other extensions to the example are multiple sellers using the same mediator, multiple parallel trades processed by
same ledger using transaction and player ids. We could have used ZK Proofs instead of Proofs of Misbehavior.

CHALLENGES AND SCOPE OF WORK

Let us look at some of the challenges with our proposed design and some areas where there are huge possibilities of improvement.

1) Global Fairness: Although we ensured two party fairness using blockchain, we still need to provide fairness guarantees at a much larger context. Looking at a particular information flow, from source to sink, at each vertex it gains value, i.e., it becomes part of a larger dataset. An individual data generator (source) is likely to claim not absolute payment from the next vertex in the flow, but a percentage of incentive from the final data consumer (sink). The percentage can be static, that is fixed during initial sale, or dynamic as the information gains value. Capturing these complicated contracts and tracking the flow for fair payout is a challenge. A bigger challenge would be disincentivizing foul play by dishonest actors and designing a protocol resilient to colluding malicious parties.

2) Efficiency: Predicate checking over encrypted data for arbitrary logic is still inefficient. Similarly, Zero-Knowledge Proofs which are non-interactive, succinct and captures any complicated generic logic are hard to develop. Proofs of Misbehavior over large circuits are still inefficient for practical purposes. There is a huge scope for development of algorithms in this space.

3) Codifying Law: Laws and regulations are written in natural languages. They can sometimes be hard to fully capture in a programming language. Even if small nuances are missed, they can lead to huge losses and data breaches. It is important to capture laws into programmable logic and verified using formal verification methods.

4) Beyond Data Trade: A pragmatic data marketplace should have features much more than just data trade, like search, analytics and computation over data. As we talk of a privacy preserving and secure portal, we need to include techniques from searchable encryption, encrypted analytics and computation over encrypted data. We should provide verifiable results or guarantees so that no party can cheat another and hence provide a fair platform. Inclusion of these added features to the framework would bring added constraints, but we need to show that the basic marketplace properties are still guaranteed.

CONCLUSION

In this paper, we have enumerated the properties desired of any practical data marketplace, outlined ways and means to design such a marketplace which facilitates trade. Primarily being an idea paper, the exact instantiation of methods and possible tools to be used are not concretely specified. The system design in its entirety is under construction. Implementation of a prototype is ongoing work through which we aim to further understand implementation issues like scalability and performance. The long term goal is to design and implement a production grade marketplace enabling data trade.
Algorithm 1: Seller and Buyer

1 func Seller {
2 Register with L as seller with public key pkS and address aS where she wants to receive money.
3 Generate key k and encrypt D using k, except columns id and Class of Disease, to get E_k(D).
4 Sign using secret key skS (ρ, pS, cS) where cS is the commission percentage it is willing to give to mediator.
5 Send E_k(D) and signed (ρ, pS, cS) to M.
6 Watch L for new buyer B and money submission by B. When done, send k to B.
7 }
8 func Buyer {
9 Register with L as seller with public key pkB and address aB where she wants to spend money from.
10 Send (Φ, pB, cB) signed using secret key skB to M.
11 Receive E_D from M and check Φ(D) = 1. Terminate if Φ not satisfied. Also receive signed p, ρ from M. Verify signature.
12 Fetch p′, ρ′ from L and check p = p′ and ρ = ρ′.
13 Send p coins (money) to L.
14 Fetch k from S and decrypt E_D using k to get D. If D does not satisfy Φ, for example there exists duplicate rows in D and hence 10000 rows of data is promised but not received, then B generates a Proof of Misbehavior and submits to L. Otherwise it sends and Ack to L.
15 }

Algorithm 2: Mediator

1 func Mediator {
2 Register with L as seller with public key pkM and address aM where she wants to receive money.
3 Receive and store encrypted data E_D. Also receive signed (ρ, pS, cS), check signature using public key of seller and if the terms pS, cS are acceptable, countersign (ρ, pS, cS) using secret key skM and send to L.
4 Receive Φ and signed commit(Φ), pB, cB from B, check signature using public key of B, pkB. Check if pB ≥ pS and cB is acceptable. Run Φ(E_D) to check if it returns 1. As E_D contains the columns id, Class of Disease unencrypted, it can easily check Φ satisfiability. If all conditions check out, countersign commit(Φ), pS, cB using skM and send to L.
5 Send E_D to B along with signed p = min(pS, pB), ρ to B.
6 }

**Author Bio**

**Banerjee, Prabal:** Prabal Banerjee joined Indian Statistical Institute as a Ph.D. student in 2016. He received his B.Sc. and M.Sc. in Computer Science from St. Xavier’s College, Kolkata and Chennai Mathematical Institute, Chennai respectively. He is currently working on integration of data with blockchain.

**Ruj, Sushmita:** Sushmita Ruj is currently an Assistant Professor at Indian Statistical Institute, Kolkata, India. Her research interests are in blockchains, IoT security, cloud security and privacy.

**References**

[1] “The worlds most valuable resource is no longer oil, but data - Regulating the internet giants,” accessed on 11-11-2018. [Online]. Available: https://www.economist.com/leaders/2017/05/06/the-worlds-most-valuable-resource-is-no-longer-oil-but-data

[2] P. Olson and A. Tilley, “The Quantified Other: Nest And Fitbit Chase A Lucrative Side Business,” accessed on 11-11-2018. [Online]. Available: https://www.forbes.com/sites/parmyolson/2014/04/17/the-quantified-other-nest-and-fitbit-chase-a-lucrative-side-business/

[3] J. B. Foster and R. W. McChesney, “Monthly Review — Surveillance Capitalism,” accessed on 11-11-2018. [Online]. Available: https://monthlyreview.org/2014/07/01/surveillance-capitalism/

[4] “WhatsApp FAQ - Backing up to Google Drive,” accessed on 11-11-2018. [Online]. Available: https://faq.whatsapp.com/en/android/28000019/?category=5245251

[5] S. Goldfeder, J. Bonneau, R. Gennaro, and A. Narayanan, “Escrow protocols for cryptocurrencies: How to buy physical goods using bitcoin,” in Lecture Notes in Computer Science, 2017, pp. 321-339.

[6] H. Subramanian, “Decentralized blockchain-based electronic marketplaces,” Commun. ACM, vol. 61, pp. 78–84, 2017.

[7] C. Serban, Y. Chen, W. Zhang, and N. H. Minsky, “The concept of decentralized and secure electronic marketplace,” Electronic Commerce Research, vol. 8, pp. 79–101, 2008.

[8] G. Lacoste, “SEMPER: A Security Framework for the Global Electronic Marketplace,” COMTEC, vol. 4, no. 4, pp. 1–11, 1997.

[9] M. T. Dashi and S. Mauw, “Fair Exchange,” in Handbook of Financial Cryptography and Security. Chapman and Hall/CRC, 2010, pp. 109–132.
Algorithm 3: Ledger Functions

```
1 func L.RegisterBuyer(pk, add) {
2     Set Buyer.pk = pk and Buyer.add = add
3 }
4 func L.RegisterSeller(pk, add) {
5     Set Seller.pk = pk and Seller.add = add
6 }
7 func L.RegisterMediator(pk, add) {
8     Set Mediator.pk = pk and Mediator.add = add
9 }
10 func L.SubmitMoney(p, sign) {
11     Verify balance(Buyer.add) ≥ p.
12     Verify sign using Buyer.pk.
13     Withdraw p coins from Buyer.add.
14 }
15 func L.SubmitAck(sign) {
16     Verify sign using Buyer.pk.
17     Send (cS + cB) × p to Mediator.add and p − (cS) × p to Seller.add.
18 }
19 func L.SubmitPoM(sign, π) {
20     Verify sign using Buyer.pk.
21     Verify Proof of Misbehavior π against commit(Φ). If incorrect, call SubmitAck().
22     Send p to Buyer.add.
23 }
```

[10] “Spotify hit with $1.6 billion copyright lawsuit — Reuters,” accessed on 11-11-2018. [Online]. Available: https://www.reuters.com/article/us-spotify-lawsuit/spotify-hit-with-1-6-billion-copyright-lawsuit-idUSKBN1ER1RX

[11] “Metrics FYE: Estimated Reach, Streaming Reactions and Graph API — Facebook Newsroom,” accessed on 11-11-2018. [Online]. Available: https://newsroom.fb.com/news/2016/12/metrics-fye-estimated-reach-streaming-reactions-and-graph-api/

[12] S. Nakamoto, “Bitcoin: A Peer-to-Peer Electronic Cash System,” Tech. Rep., accessed on 11-11-2018. [Online]. Available: www.bitcoin.org/bitcoin.pdf

[13] Z. Kakushadze and R. P. Russo, “Blockchain: Data malls, coin economies, and keyless payments,” The Journal of Alternative Investments, vol. 21, no. 1, pp. 8–16, 2018.

[14] I. Bentov and R. Kumaresan, “How to Use Bitcoin to Design Fair Protocols,” in CRYPTO, 2014.

[15] A. R. Choudhuri, M. Green, A. Jain, G. Kaptchuk, and I. Miers, “Fairness in an Unfair World: Fair Multiparty Computation from Public Bulletin Boards,” ACM CCS, pp. 719–728, 2017.

[16] N. Herbaut and N. Negrini, “A Model for Collaborative Blockchain-Based Video Delivery Relying on Advanced Network Services Chains,” IEEE Communications Magazine, vol. 55, pp. 70–76, 2017.

[17] “Zero Knowledge Contingent Payment - Bitcoin Wiki,” accessed on 11-11-2018. [Online]. Available: https://en.bitcoin.it/wiki/Zero_Knowledge_Contingent_Payment

[18] S. Delgado Segura, C. Prez-Sol, G. Navarro-Arribas, and J. Herrera-Joancomart, “A Fair Protocol for Data Trading Based on Bitcoin Transactions,” Future Generation Computer Systems, 2017.

[19] S. Dziembowski, L. Eckey, and S. Faust, “FairSwap: How to fairly exchange digital goods,” in ACM CCS, 2018, pp. 967–984.

[20] D. Sønstebø, “IOTA Data Marketplace IOTA,” accessed on 11-11-2018. [Online]. Available: https://blog.iota.org/iota-data-marketplace-cb68c465ac70

[21] “Decentralized Data Marketplace (DDM),” accessed on 11-11-2018. [Online]. Available: https://github.com/ANRGUSC/DDM

[22] Fysical Technologies Pte. Ltd., “Fysical: A Decentralized Location Data Market,” Tech. Rep., 2018, accessed on 11-11-2018. [Online]. Available: https://fysical.org/tokensale.html#papers-section

[23] “Streamr,” Tech. Rep., accessed on 11-11-2018. [Online]. Available: https://s3.amazonaws.com/streamr-public/streamr-token-whitepaper-2017-07-25-v1_0.pdf

[24] P. Missier, S. Bajoudah, A. Capossele, A. Gaglione, and M. Nati, “Mind my value: A decentralized infrastructure for fair and trusted IoT data trading,” in Proceedings of the Seventh International Conference on the Internet of Things, 2017, pp. 15:1–15:8.

[25] T.-D. Cao, T.-V. Pham, Q.-H. Vu, H.-L. Truong, D.-H. Le, and S. Dustdar, “Marsa: A marketplace for realtime human sensing data,” ACM Trans. Internet Technol., vol. 16, no. 3, pp. 16:1–16:21, May 2016.