1
Introduction

The development of the personal computer and the internet as the digital infrastructure for global connectivity drastically impacted almost every feature of our lives. Many cannot imagine life without constant global acccess to knowledge, photos, social contacts and many more things. When Tim Berners-Lee developed the World Wide Web in the 1980s he imagined it as a globally distributed source of knowledge in form of a network where each node is equal, can both contribute and consume knowledge in order to improve cooperation and sharing of information. The huge success of the internet proved his idea right but at the same time led to problems: with global connectivity the openness of the internet's network invited malicious users. Also some services offered on the web were highly requested which required a highly advanced infrastructure to handle such traffic. Reasons such as these led to the centralization of the internet and the monopolization of services.

A similar trend can be found in many aspects of modern society: global production has replaced local production, global retailers replaced local retailers and mega cities replace rural villages. The strive for efficiency, growth and success leads to entities combining forces and creating ever stronger super-entities that improve processes to become more powerful. Mostly centralization has led to widespread welfare through more jobs, more affordable products and more efficient distribution of goods and services. The same is true for the internet: facebook has connected more than 2 billion people, google allows you to find information on anything and youtube is delivering entertainment; all for free, available to anyone willing to create an account, world wide.

But not everything about centralization is advanatgeous. Companies in powerful monopoly-like market positions can misuse their importance to influence politicians to act in their favor. German automobile manufacturers for example lobby against policies for reducing emissions of cars in European Parliament or speed-limits on the German Autobahn. In the digital world, Facebook misuses their user's data and allows political propaganda to be spread to millions of people. Furthermore, centralized application are prone to attacks by hackers as the servers are a single point of failure, and one data breach leads to the exposure of millions or even billions of users. Many such examples have proven in the past that centralized applications are not safe.

Banks are another example of such centralized systems that have their client's trust but are proven untrustworthy like during the financial crisis. This led to the development of Bitcoin, the first digital, decentralized currency which works completely without central authorities and is safe against double-spending, an attack on digital financial systems in which two confilcting transactions are submitted, resulting in spending some amount of currency twice. Bitcoin uses a single global hash chain of blocks (sets) of transactions. The hash chain leads to a chronology of transactions which makes double-spending impossible. All nodes on the Bitcoin network agree on this sequence of transactions using an algorithm called Proof-of-work(PoW) which solves a hard mathematical problem, expending time and resources. The first to solve the problem can publish a new block on the chain including a new (confirmed) set of transactions. Bitcoin is very secure, completely decentralized but it's Achilles heel is scalability: with a single global chain of transactions and the current block time of 10 minutes the theoretical throughput is 7 transactions per second.

Since the advent of Bitcoin many other digital currencies have appeard and some have found almost similar praise as Bitcoin, however the scalability problem still exists with most of the currencies. Vitalik Buterin, co-founder of the second largest digital currency network Ethereum, describes this as a trilemma: of the three desireable properties decentralization, scalability and security, at most two can be attained by one blockchain system at the same time. But in order to be usable for a global currency or as a global transaction storage, scalability will be neccessary.

This master thesis is concerned with TrustChain, a blockchain system developed at the Blockchain Lab at TU Delft. Instead of designing the system as a security-first system, TrustChain focusses on scalability. TrustChain has no global consensus on all global transactions which removes the bottleneck for transaction throughput. That means, that there is no single blockchain but each entity on the network has their own blockchain. We have shown in previous work that this system achieves horizontal scalability, so additional nodes on the network lead to additional throughput. The scalability comes at the cost of security guarantees, most prominently the double spending attack and the sybil attack. The double spending attack is easy to perform because an agent can simply publish a second conflicting transaction with another node without sharing the original transaction. An attack is called a Sybil attack when an adversary creates many fake entities on the network to obtain a large part of the voting power or resources. This attack is usually prevented through verification of the agent, for example bitcoin verifies nodes by letting them perform work. Trustchain is built in a way that these attacks are not prevented, which is too costly to be scalable, but instead are detectable. Both transactions of a double-spend attack are stored on the blockchains of the two exploited agents. By spreading the records of these transactions in the network, eventually a node will have both versions of the same transaction and will identify the attacking node.

Our contribution will be targeted at the dissemination mechanism of transaction records on the Trustchain network. Because every node has it's own chain, each node has only a subset of the data. This is different from blockchain systems with global consensus and therefore is an under explored topic. While Trustchain is made in a way that makes attacks detectable this is only possible if an agent collects data from other agents on the network. This mechanism has not yet been formerly been defined and it's implications for security and scalability of the Trustchain have not been researched in depth.

Next to the research on Trustchain the Blockchain Lab is concerned with real-world testing of a deployed system. The platform for those tests is Tribler, an anonymized onion-routing protected bittorrent client. A common problem with the bittorrent network is that it is not inherently protected against free-riders as there is a social dilemma in sharing resources with other agents on the network. Uploading data to other nodes is costly in terms of bandwidth without any direct reward, but if noone decides to upload nobody can download and the system breaks. This dilemma is a form of what is known as the Prisoner's dilemma in classical game theory. With BarterCast our research group has devleoped a system to prevent free-riding in the bittorrent network and recently this mechanism has been improved using Trustchain as a tamper-proof way to store transaction records. This work will also use the Tribler example as context.

The rest of this report will be structured as follows. In the next chapter, the problem will be discussed in more detail.

2
Problem description

Any reputation system requires a few basic properties in order to be useful for building trust among strangers. According to Resnick et al. [1] those are:

- long-lived entities
- feedback about current interactions is captured and distributed.
- past feedback guides decisions

Reputation systems have been shown to work well on online centralized marketplaces such as Ebay, Amazon or home sharing service Airbnb. The above properties are quite easy to fulfill in a centralized system. Long-lived entities can be enforced through requiring document checks on sign-up, requiring real names and invalidating email adresses after use. Feedback of interactions can be stored in databases, distribution is then as simple as connecting a rest-api to the database. With access to past feedback on interactions users can make informed decisions on whom to trust.

In decentralized reputation systems those properties are harder to fullfil and at the same time other constraints play a role, like scalability and manipulation resistance. With TrustChain our research group has developed a scalable solution to record feedback of past transactions in a distributed fashion. Furthermore we have developed Sybil-resistant accounting policies to make decisions based on that feedback. With the availability and improvement of biometric authentication on common mobile devices long-lived entities will be a solved problem.

While we are able to store past interactions in the multichain storage, the complete information of the network is fragmented and each agent only has information on its own transactions. This leads to security problems as many attacks are only detectable if certain information from multiple agents is combined. An example of this is the double spending attack in which two conflicting blocks with equal sequence number are created and sent to two different agents. Only if another agent has access to both blocks can the attack be detected. This requires a dissemination mechanism, a way for nodes to share data with each other. For Trustchain such a mechanism has not formally been defined. Not only the detection of attacks is dependent on the available information, but also the allocation of resources. In a previous work we defined multiple allocation policies which take the subgraph of the network as input and output a distribution of resources for requesting agents. The allocation policy should reward good network reputation and punish bad reputation, but the reputation is dependent on the knowledge of past interactions. With more available information the agent's approximation of other agents' reputations becomes more accurate.

2.1. Research question

From the above discussion we can formulate the following research question:

How can we design a transaction record dissemination mechanism such that information is widely distributed, tamper-proof and sharing of information is strategy-proof?

3
Related work

3.1. Context

To understand the relevance of this work we need to put it into the perspective of the context of reputation systems and their applications. With the wide-spread use of the internet for trade, sharing and communicating, interactions between strangers living far apart are wide-spread. Many applications allow for exploitation through manipulation and taking advantage of the asymmetry of information. For example a buyer needs to pay for products before even seeing and afer receiving the money the seller actually does not have any incentive to deliver a product. This opens the door for exploitation. In order to solve this problem a reputation system is put into place in ebay, which shows publicly what other buyers said about the seller in the past. If the seller has not delivered a product a few times, the reputation will tainted with negative reviews and buyers will be reluctant to interact with this seller in the future. Also, having no reputation at all will seem suspicious and buyers will have little trust in the seller. This influences the prices of products that this seller can ask for.

3.1.1. Reputation systems

Reputation is a concept that not only exists on internet platforms, but it is an important part of everyday life. Everyone has opinions about friends, colleagues, companies, newspapers, weather forecasts and many more things. Reputation and trust are subjective quantities, we are influenced by gossip from our peers. If all of our friends tell us that a certain company makes crappy laptops, we are probably choosing for a different companies laptop.

In this work we will focus on reputation systems for internet platforms. In this category there are two different approaches, the centralized approach and the decentralized approach.

Centralized ...

Decentralized ...

3.1.2. Applications

Market

Sharing economy

BitTorrent

Tribler

3.2. TrustChain

3.2.1. Data structure

3.2.2. Accounting mechanism

Definition of trust and reputation

3.2.3. Subjective graph

3.2.4. Consensus

4

Attacks

5 Information dissemination

The strength of a reputation system largely pivots on the availability of data. Reputation is built through a history of interactions, but only those that know about the history can estimate the true nature of the agent. Also, unfair actions can only be detected if the information about those actions is widespread. In centralized reputation systems, this availability in guaranteed, as long as the central server with all data is available and not manipulated. In decentralized systems this guarantee does not exist, availability of data depends on the willingness of agents to share their private data.

Therefore our goal is to create a mechanism that gives agents an incentive to share their private data. More specifically, this dissemination mechanism should make the sharing of private data strategyproof, that is, sharing all private data should never be less advantageous than not sharing.

5.1. Pairwise auditing

agents group in pairs and assign a score(endorsement) to each other. The endorsement should increase with more data shared between the two parties. There is a maximum endorsement which can be calculated when all data is available. The score is used as another factor when determining the trustworthiness of an agent.

5.1.1. Definition

5.1.2. Incentive design

Endorsements without any endorsements, agents are seen as not trustworthy. Therefore agents need to exchange data with at least a few agents in order to become trustworthy. Endorsements should not be accepted by default but rather only be accepted

Strategy-proofness

5.1.3. Endorsement

5.2. Accusation

5.2.1. Proof

5.2.2. Untrue accusations

6

Experiments and results

6.1. Setup

6.1.1. Dataset

6.1.2. Experiments

6.2. Valid usage

More data leads to higher score

More audits, leads to higher average trust

6.3. Sybil attack

Sybil vs true agent

Detectability of lying

6.4. Accusations

7
Discussion

7.0.1. Strategy proofness

7.0.2. Attack resistance

7.0.3. Future research

Bibliography

[1] Paul Resnick, Ko Kuwabara, Richard Zeckhauser, and Eric Friedman. Reputation systems. *Communications of the ACM*, 43(12):45–48, 2000.