DECENTRALIZED REPUTATION MODEL AND GENERAL TRUST FRAMEWORK

Dissertation in partial fulfillment of the requirements for the degree of

MASTER PROGRAMME IN COMPUTER SCIENCE

UPPSALA
UNIVERSITET

Uppsala University
Department of Information Technology

SUJATA TAMANG

March 14, 2018

DECENTRALIZED REPUTATION MODEL AND GENERAL TRUST FRAMEWORK

Dissertation in partial fulfillment of the requirements for the degree of

MASTER PROGRAMME IN COMPUTER SCIENCE

Uppsala University
Department of Information Technology

Approved by

Supervisor,   Jonatan Bergquist

Reviewer,   Björn Victor

Examiner,   Prof. fName lName

March 14, 2018

Abstract

Abstract here

Acknowledgements

Acknowledgements here

Table of Contents

List of Tables

List of Figures

List of Abbreviations

1 Introduction

1.1 Definition

1.1.1 Trust and Reputation

Trust and Reputation encompass a wide spectrum of domains and is context dependent. Therefore, a single definition that is universally agreed upon doesn't exist. If we take the game theoretic approach of viewing them, trust can be defined as a subjective probability, by which an individual, A expects another individual, B, to perform a given action on which its welfare depends according to a previous agreement. Similarly, reputation can be defined as the perception of an individuals character or standing **Sabater2005**. Individuals in online systems are identified by their online identities which can be anything and not necessarily attached to real-world identities. Online identities play a crucial role in digital interactions and require unknown entities to trust each other based on the reputation system of the platform in use. The client-server architecture of most online systems makes it easy to attack them. For instance, performing a DOS attack would require attacking one central server and all the users whose information lives on that server will lose access during the time of the attack. The architecture with a central point of authority has a centralized entity in control of the database. While users may be able to read or modify data, it is the central authority that one needs to rely on for providing trustworthy information enough to make a good decision on the network.

1.1.2 Blockchain

Blockchain can be defined as a distributed record of state changes that lets anybody on the network audit state changes and prove with mathematical certainty that the transactions were made according to the blockchain rules **enoughBitcoinForEthereum**. Each block consists of lists of validated transactions and are linked by cryptographic hash to previous block. Thus, it is practically impossible to modify a block without being evident on the network. This technology came into being in 2008 as Bitcoin, a peer-to-peer electronic cash system, **Bitcoin_Satoshi** which allowed unknown and not necessarily trusted entities to transfer value to each other without the need of any central trusted third party. Blockchain can be permissionless or permissioned, public or private depending on the application and use cases. Different applications may employ different consensus algorithms to maintain and update the blockchain database. Blockchain ensures fault tolerant, zero downtime, and tamper resistant data to be stored on its network.

1.2 Motivation

Any form of digital interaction that takes place online requires participating entities to trust each other based on reputation system of platform in use. Let us consider a simple scenario where Alice wants to buy a pair of headphones for which she browses a buy/sell platform. She then finds a relevant ad placed by Bob who has good reviews and rating in the respective platform. Alice picks up her credit card, submits required details and waits for her desired product to be delivered as promised. In this interaction, trustworthiness of Bob is fully reliant on the reputation/rating system of the platform and the correctness of information stored on their database. The entity claiming to be Bob could be Eve who found a way to bypass the platform's security and inflate his reputation on the system. Eve could delete the ad and associated account when the payment is complete or she could gather Alice's personal details to misuse it later. Any malformed decision on trustworthiness of an entity could be expensive and deal severe damage to the user. Thus, it is interesting to study about reputation systems and methods to make it more reliable and accurate in its measure. We can say reputation system offers a way to measure trustworthiness of entities to aid interacting entities in making an informed decision about carrying forward the transaction or dropping it. Studying interactions between entities and analyzing their behaviour to generalize a trust framework is therefore a riveting problem. Graph theory and network flow algorithms have been used in the past (cite) to study interaction between nodes and prevent any possible malicious behaviour in the network. Blockchain is a fairly new technology and various use cases are being proposed and developed. To name a few of them, Sovrin(cite) is a blockchain based solution for self sovereign identity and decentralized trust. Wepower(cite) is another innovative use case that aims to tokenize the renewable energy. Cardano is a blockchain platform that uses a ouroboros, proof-of-stake consensus algorithm (cite). Similarly, policypal(cite), uport(cite) are other blockchain based application focused on diverse use cases. Leveraging blockchain technology to implement reputation system could be an ideal solution for measuring trustworthiness and attempt to increase accuracy of trust.

1.3 Purpose and research questions

The main goal of this thesis is to use blockchain technology and smart contracts to simulate an endorsement network where entities can endorse each other based on physical or digital acquaintance. The endorsement values will be quantified to infer reputation score and a trust value that can be used on any transaction network. The nodes and their relationship will be studied to identify honest or malicious participants. Generalization of this endorsement network to serve other use cases will also be discussed. The research questions that this thesis aims to address are :

1. How can graph theories and relevant reputation algorithms be used to model the interaction between entities and detect/identify honest and malicious nodes in the network? How can the interaction graph be modeled?

2. What are the requirements for storing trust values and linking them to associated identities stored off a blockchain network? How can a blockchain application be built to define a general trust framework for a transactional network? How could the overall system architecture look like?

3. How can the discussed endorsement network ensure trustworthiness while also preserving users anonymity and how can it be generalized to other transactional network or added on top of it to serve other use cases such as content filtering, E-Commerce etc?

1.4 Scope

This thesis work attempts to answer all the research questions mentioned in section(refer section).

For Research question1, literature survey will be performed on various reputation algorithms in use and a background overview will be performed that will lead to graph simulation of endorsement network. For Research question2, methods to quantify reputation scores and trust values will be devised. Blockchain storage and off blockchain storage such as ipfs will be studied and both methods will be compared and analysed. The system architecture for proposed trust framework will be presented. For Research question3, Several use cases will be presented and the network will be tested on with various predefined cases to see how well it acts in the dynamic network.

1.5 Structure of Report

1.5.1 acronyms

2 Literature Review

2.1 Existing Reputation Systems

2.1.1 Centralized

Currently, most of the reputation models deal with users feedback after they have completed a transaction in their online platform of choice. Popular auction site, ebay(footnote) is widely believed to be the biggest and best known for its reputation model. It uses a transaction based system where users are able to leave feedback about the interaction they had and can leave ratings as 0(neutral), 1(positive) or -1(negative). These values are aggregated to compute a net score of users trustworthiness. Similarly, Q&A websites such as Quora(footnote), Stackexchange(footnote) have their own proprietary reputation model to calculate the users standings in their network. Almost all the methods used are explained in web of trust(cite paper) where badges, karma, activity level of users are brought into play. The common issue with all of them are that they are centralized system with central point of authority and thus lacks transparency in correctness of information displayed. i.e. It is hard to tell if the information displayed about a users standing is really the aggregated total of received feedback from honest users or there were human intervention in the process. As it is a centralized system, it is certainly possible for the admin/mod to change the rating for their favored user. Other possibilities include an attacker getting access to the system and changing the values to match their need.

2.1.2 Decentralized

There have also been various studies and implementations for decentralized systems such as BitTorrent, gnutella network to employ an efficient reputation system. Tribler(cite) uses blockchain based solution to keep track of users activity in the network. Similarly, Beaver(cite) is a decentralized anonymous marketplace where reputation scores and trust values are inferred from the participating users feedback but they are left completely anonymous. TrustDavis(cite) presents a solution for non exploitable online reputation system by defining important characteristics of honest and malicious participants and incentivizing accurate ratings provided by the user and discouraging dishonest behaviour.

2.2 Problems & Limitations

The major problem with centralized system is their central point of authority. Therefore, it is hard to always trust the ratings shown there. Since, a human is in control of the database, they might change it to meet their need or an attacker can gain access to it and manipulate the system to meet their need. Therefore, they cannot assure a tamper resistant, always available and verifiable data with their architecture. The problem with decentralized system is Sybil attack where a user creates multiple pseudonymous identities to exploit the system and inflate his reputation. This is usually difficult to detect and have been proven to be practically impossible to prevent in a distributed computing environment (cite). Another issue is shilling attack where a user intentionally provides dishonest feedback for a transaction. There is also a possibility for multiple users to collude and manipulate the system to inflate the group members reputation.

2.2.1 Sybil Attack

Sybil attack is a widely used attack model in the peer-to-peer reputation system. Peers in the network create multiple pseudonymous identities with a purpose of inflating their reputation or damaging some other peers reputation. If a peer gets a bad reputation in the system for its activity or other reputation models defined parameters, then usually it is both cheaper and faster to create a new identity and start afresh then to try and recuperate the damaged reputation. As the network makes it so easy to create identities with nothing at stake, participants opt for it and exploit this feature to perform Sybil attack.

3 Background

3.1 Reputation algorithms

3.1.1 Graph properties

A graph, as the name suggests can be used to represent objects and their relationships graphically. A graph G is an ordered triple (V,E,φ_G) where V is a non empty set of vertices v, E is a non empty set of edges e that connects two vertices and $v \in V, e \in E$. φ_G is an incidence function that assigns pair of vertices to each edge of the graph G. φ_G(e) = uv represents that e is an edge that joins vertices u and v. Graph properties can be leveraged to represent an interaction graph of network for reputation system. Each node on the network, v can represent individuals and the edges that connect the nodes can represent the relationships between those nodes. The edge can have varying weights to represent the strength of relationship between the nodes. **bondy1976graph**

3.1.2 EigenTrust

EigenTrust is a reputation management algorithm for P2P network that aims to minimize malicious behaviour in the network and is based on the notion of transitive trust. i.e. If a peer *i* trusts a peer *j* then all other peers trusted by *j* is also trusted by *i*. In EigenTrust, global reputation of each peer *i* is given by local trust value assigned to peer *i* by other peers and is weighted by the global reputation of assigning peers. A local trust value s_{ij} is calculated by each peer *i* which represents the opinion *i* has of *j*. s_{ij} is the difference of satisfactory and unsatisfactory transactions peer *i* had with other peers *j*.

$$s_{ij} = sat(i,j) - unsat(i,j) \tag{3.1}$$

where sat(i,j) represents number of satisfactory transactions that *i* had with *j* whereas unsat(i,j) represents number of unsatisfactory transactions.
To prevent malicious peers from assigning arbitrarily high local trust values to other malicious peers, the local trust value is normalized as c_{ij} before aggregating them.

$$c_{ij} = \frac{max(s_{ij},0)}{\sum_j max(s_{ij},0)} \tag{3.2}$$

C_{ij} keeps changing depending on the good or bad interaction between peer *i* and peer *j*. Based on the local trust value assigned by other peers, each peer has a global trust value that determines their standing in the network. To aggregate the normalized local

trust values, the approach used is friend-friend reference where a peer i would ask its acquaintances about their opinion about other peers. Trust that peer i places in peer k by asking his friends can be denoted by t_{ik} as :

$$t_{ik} = \sum_j c_{ij} c_{jk} \tag{3.3}$$

Each peer asks other peers about their opinion which is weighted based on how much peer i trusts them. If we define C as a matrix $[c_{ij}]$ and t_i as a vector containing values t_{ik}, then $t_{ik} = C^T \vec{c_i}$. This helps a peer get a wider view of the network more than its own experience. This can continue for many nodes until peer i asks his friend's friend's and friend's friend can be consulted further to receive a broader view of the network. For n nodes, we can represent t as t = $(C^T)^n c_i$. For a large enough value of n , trust vector $\vec{t_i}$ will converge to same vector for every peer i and could give complete view of the network. t is the global trust vector where t_j quantifies the trust system places in peer j. EigenTrust is robust to malicious peers and good for decreasing inauthentic file downloads in a P2P network. However, it doesn't address the issues such as inactive peers, where a peer doesn't download from anywhere else, malicious collectiveness, where malicious peers collude to inflate the trust value. It also doesn't have a way to calculate negative trust and is entirely based on user feedback. **kamvar2003eigentrust**

3.1.3 Net flow Rate convergence

Net flow rate convergence can help to determine anomaly in the network. By looking at how fast the net flow converges to zero, it can detect unusual behaviour in the network. The flow in a network can be measured by looking at inflow and outflow edges and calculating their differences. Inflow edges are all incoming edges in the graph and outflow edges are all outgoing edges . (diagram) Net flow convergence rate is the rate at which the net flow converges to the global net flow which is zero. Depending upon how fast the net flow in a graph converges to zero, it can be useful to detect anomaly.(example diagram)

3.1.4 Binomial Random Walk

3.2 Cryptography

3.2.1 Basic Concepts

Cryptography offers algorithms to achieve confidentiality, integrity, authenticity and non repudiation. Confidentiality and integrity ensures that the information being communicated is not disclosed or has been modified to or by any unauthorized parties. The data is hidden or encrypted such that only the authorized parties can make sense out of it i.e. decrypt using the previously agreed upon key. An illustration of sending encrypted messages that can only be decrypted with the shared key is presented in

figure 3.1 Authenticity and non repudiation are another properties that makes sure
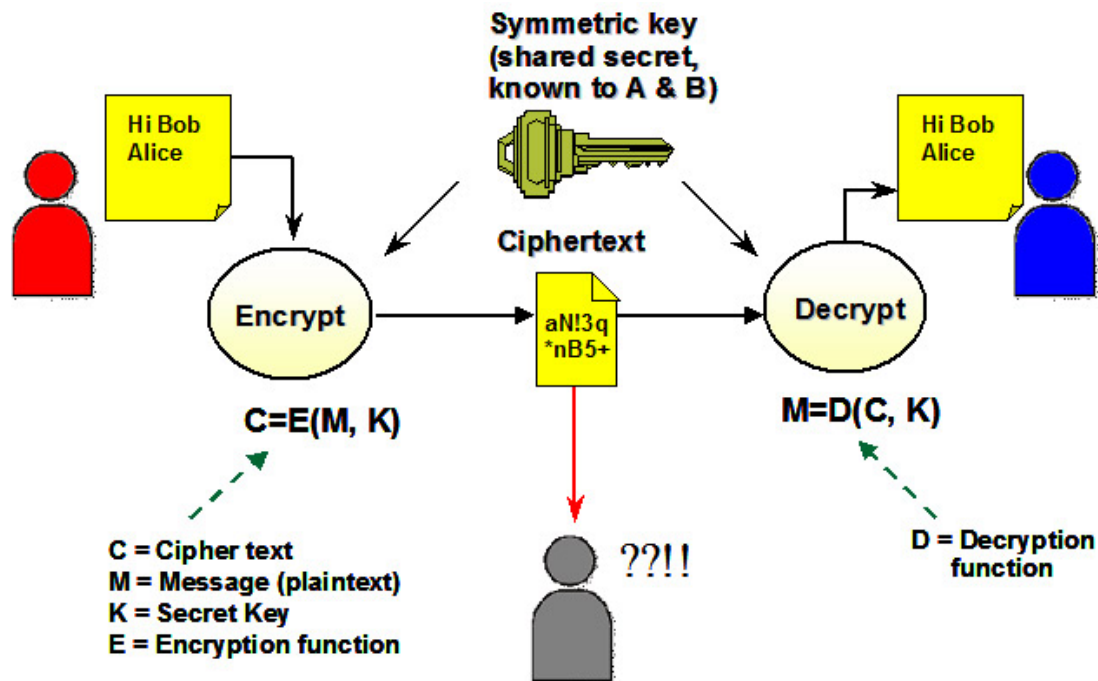
Figure 3.1: Symmetric Key in use **symmetrickey**

that the sender is who he claims to be and that the sender cannot deny having sent the message. For these properties, symmetric key cryptography wouldn't be enough. This is where Asymmetrical cryptography or public key cryptography comes in which uses key pairs, private key known only to the owner and public key that can be distributed publicly. Public key verifies the holder of the private key and encryption of message. This encrypted message can only be decrypted by that paired private key. One of the major application of public key cryptography is Digital Signatures, described in more detail in section (Blockchain section ..) which is useful in preserving the properties of authenticity and non repudiation.

A cryptosystem can be seen as a five tuple (P,C,K,E,D) that satisfies the following conditions:

P is a finite set of plain texts.

C is a finite set of cipher texts.

K, the keyspace is a finite set of keys

E, set of encryption rules e_k: P $\Rightarrow$ C

D, sete of decrytion rules d_k : C $\Rightarrow$ M.

for each k $\in$ K, there is $e_k \in$ E and $d_k \in$ D such that $d_k(e_k(m))$ = m for every plaintext m $\in$ P.

3.2.2 Hash functions

Cryptographic hash functions are one way function, also known as mathematical trap door function that transforms an input message into a fixed length binary output. It is one way because although transforming a message input to a hash value or a message digest can be done in constant time, reversing the operation is practically impossible to achieve as its computationally inefficient. Earlier hash functions include MD5 which produces a 128 bit hash value but is vulnerable and can be cracked by brute force attack. The predecessors hash functions are sha-256 preceeded by sha-1, sha-2 and others. Their applications include digital signature, message authentication both of which are interesting for blockchain as will be discussed in section(name). The important characteristics of hash functions are their deterministic output, meaning given a fixed input, it will always generate the same output. It offers collision resistant property i.e. it is impossible or extremely rare to get the same hash value for two different messages. If m1 and m2 are the message and h(m1) and h(m2) are hash functions applied to them respectively, collision resistant ensures that h(m1) != h(m2). Another important characteristic of hash function is that the hash value provides no indication of the original information that was hashed thus making it efficient for hiding information.

3.2.3 Digital Signature

Digital signatures

3.3 Blockchain Technology

3.3.1 Basic Concepts

transaction, message, signature, broadcast, verify, network, miners, validate, write blocks

3.3.2 Evolution & Categories

Bitcoin,Ethereum, private permissioned

3.3.3 Consensus algorithms

proof of work, proof of stake, proof of validatioin, proof of authority, and others

3.3.4 Smart contracts

define,explain

3.3.5 Applications

4 Methodology and Implementation

4.1 Problem Statement

The problem of measuring the trustworthiness of communicating entities is an essential aspect of any online system where entities interact with each other for any purpose, be it shopping, content delivery or file sharing. An endorsement network where physically or digitally acquainted entities can endorse each others presented information is proposed. This network comes with several questions such as who are the endorser and endorsee, why would they endorse the correct information, what would stop someone from just endorsing a random piece of information or creating multiple identities and endorsing themselves, etc. The storage and aggregation of information and their computation to generate a trust value for each participating node will be discussed as a blockchain based solution in the sections further.

4.2 User stories & Requirements

The two primary users in the model described above is endorser and endorsee where endorser can initiate a transaction by giving the endorsement to an entity which he trusts, and endorsee can accept the transaction that was initiated by the endorser to receive more points in the system. The requirements can be listed in points as :

- Anyone should be able to join the network.

- There must be a way to be able to associate information presented in the network to the unique id of a user and assign the relevant score to it.

- Endorser should be able to broadcast their intention of giving an endorsement to the endorsee and securely sign the transaction in a way that identifies them such that other participants in the network can verify the transaction.

- Endorser should be able to take back the endorsement that was previously transferred in event of which the whole chain on network that are connected to that particular transaction will have the previously assigned scores eliminated.

- Endorsee should be able to accept the endorsement given to them by the endorsers and be able to view the endorsers information that is permitted by the endorser.

- Both endorser and endorsee should be able to view the trust scores associated with their unique id.

- All the successfully recorded transactions should have an immutable traceability such that anyone can go back and verify the chain of ownership and order in time when it actually took place.

- Any form of attempt to change the successfully recorded ledger in block should be evident on the network, if not resistant.

The functional requirement for each user types specifically are presented in the table(cite) The non-functional requirements are associated with the architecture and are of interest to the network participants:

- Security:

- Reliability:

4.3 Component Diagram

4.4 The Model - Endorsement Network

The initial assumption is that all nodes are honest, thereby all nodes that join the network are given an initial endorsement power of 1. The value is 1 to avoid misuse of high values later on in the game by a single node to just inflate their impact. This value keeps getting diluted with each interaction of given endorsement. This process will be further clarified later with an example.

Terminologies that will be used for this network are briefly described below :

nEG: The number of connection to whom a peer has given endorsement.

nER: The number of connection from whom a peer has received endorsement.

Initial Endorsement Power (iep): This is the initial endorsement power granted by the network for being a participating node in the network.

itEndorsement power (ep): Endorsement power is measured by how much points has been given by the endorser which can be determined by the number of connections. If a peer i uses his *iep* to give to n number of peers, then the ep_i is $1/n$. For instance, if A endorses 20 acquaintances, then ep_a will be $1/20$, if A endorses 50 acquaintances, ep_a will be $1/50$ and so on.

Endorsement impact (ei): Endorsement impact is not only associated with how much endorsement an entity has given but also with how much it has received. The ratio of endorsement given (EG) and of endorsement received (ER) has to be taken into account to create a balance in the network for each nodes. Assume the EG:ER is x and y respec-

tively, let total value of received endorsement be RE , the ei can be calculated as

$$ei = \frac{min(x,y)}{max(x,y)} * ep * RE \tag{4.1}$$

This is to ensure that EG and ER and not too far off from each other.

Received endorsement, RE is the total sum of all the endorsement received. If a peer receives endorsement from n peers, then the RE is given by:

$$RE = \sum_{i=0}^{n} ep_i \tag{4.2}$$

Total endorsement impact(tei): The total endorsement impact determines the impact a node has on the network. To get a value for this, we would simply have to multiply the *ei* with the number of connections to which they have given endorsement. Assuming a peer *i* has given endorsement to *n* peers in the network, then the *tei* would be:

$$tei = ei * n \tag{4.3}$$

It shows how much impact they have made on the network. This value corresponds to the trust score and higher the score, higher is the trustworthiness of an entity.

4.5 Honest vs. Malicious Nodes

In endorsement network, honest nodes are assumed to endorse only the nodes on whom they have full confidence that they will perform the fair/legitimate action on the system. In other words, they are ready to take the risk if the identities they trusted performs malicious activity. It takes time to build a reputation and gain enough trust value and therefore giving it all up by endorsing malicious node would not be a rational decision. Another assumption is that an honest node will have a negligibly low difference between nEG and nER. On the other hand, a malicious node will have imbalanced ration between nEG and nER. Using the ratio of nEG and nER as one of the metrics may also alleviate the common free rider problem discussed earlier. kkjkj

4.6 Trust Metrics

4.7 Token and Incentive

4.8 Design of PoC

4.9 Sequence Diagram

4.10 Smart contracts

4.11 Experimental Setup

4.12 second section

It may include: Description of the methodological, theoretical, conceptual or empirical framework; design of the experiment; relevant steps of reasoning; data description and sources.
Describe the approach and method(s) used to address the scientific problem. Also reflect on the particular choice of method and justify it.

5 Results

5.1 Interaction graph

5.2 Analysis

5.3 Measurement

5.4 Comparison

Presentation of results and case-study data An application of the methodology is unfolded and results are presented using for example via Charts, Diagrams, Figures and Tables The work is conducted in accordance with the method described earlier. Results are presented in an analytical way.

6 Discussion & Analysis

6.1 Generalization

6.2 first section

The results presented in Chapter 4 are discussed and analyzed, including comments and reflections from the author. It may include the following: Comparison of obtained results with discussion,interpretation and evaluation of results. Results of analysis or modeling are described. Interpretations are drawn and connected to previous work

7 Conclusion

7.1 first section

7.1.1 first subsection

Synopsis of findings, limitations, further proposals for future work on the subject. Clear conclusions are drawn that stem from the previous analysis. Present the conclusions drawn and the evidence and arguments that support the conclusions.

Do not include new findings, but only refer to results already discussed in the thesis. Relevant further work in the field is summarized.