Self-contained, trusted compensation system for community mesh networks (this title is subject to change)

January 23, 2019

Abstract

The title of 'Executive Summary' could also be used instead of 'Abstract'. The purpose of this report is to summarize my experience of working at the case study of automating a trusted and distributed compensation system in Guifi.net's community mesh network in Barcelona, Spain. The experience involves both hands-on work—in terms of being involved in programming a set of smart contracts—, literature review to have a broader understanding of the issue, and meetings and discussions with the team of people involved in maintaining the system of Guifi.net. My featured learning experiences involve learning the concept of Oracles and how to write a set of smart contracts that realizes this concept that work in harmony with the more traditional functions of vintage blockchain. Other learning experience is to get myself familiarized with the thought process related to designing a self-contained economic model for the compensation system of the mesh community network of Guifi.net. The report attempts to explain the architecture of the local experimental setup with a description of its different components along with interactions among them. Since the work mentioned in this report is ongoing, it is only apt to close it with a discussion on future directions at the end of the report.

Contents

List of Figures

1 Introduction

This section will cover the following main things.

1.1 Case study: Guifi.net

Here goes the brief introduction of Guifi.net's mesh community network. It will contain details about different facts and stats related to this network[1]. Text before this will be removed/edited.

Community mesh networks are mostly realized by a volunteer effort of a local community, e.g., comprising local citizens and organizations, and act as alternative or a complementary means to connect people, specially from areas with no to intermittent connectivity, to the Internet and its services.

Community networks' philosophy is mostly underpinned by *commons model* that entail pooling and coordination of resources and efforts of a local community to build a system. The distinguishing features of such projects are usually their openness, costlessness, and neutrality [1].

One such community network is Guifi.net which is claimed to be the largest of such networks on the planet [1]. Guifi.net's community network infrastructure comprises some more than $30,000$ nodes and 4Gpbs traffic volume[2].

1.2 Guifi.net's compensation system

Things to consider:

- Here the description of the status quo of Guifi.net's compensation system will go. Roger's paper will be briefly explained in this subsection [1].
- Consider explaining what is commons-model and why do we still need a compensation system? And we exactly are going to be the beneficiaries of this system?
- Main focus will be on the limitations of the current system and the aim is to gradually provide motivation for a trusted, transparent and distributed compensation system.
- By doing this we will eventually loop in blockchain's immutable record keeping along with a set of smart contracts that will eventually automate the compensation system for this community network.
- You can also cite the recent paper here [2].
- Skim through [3] to learn about Guifi.net's governance model.

[1] `http://guifi.net/en/what_is_guifinet`
[2] `http://guifi.net/en/what_is_guifinet`

Text before this will be removed/edited.

Baig et al. in [1] describe that sustainability and scalability are the main challenges in the domain of community mesh networks. These are particularly important mainly because such networks are mostly a result of voluntary and non-refundable contributions. Such contributions can be in terms of providing actual monetary capital or can take the form of technical services or resources required for the underlying communication infrastructure.

1.3 Limitations of the current system

The purpose here is to highlight the short comings and limitations of the current compensation system. This can be done in terms of describing trust issues when it comes to manual record keeping. We can also describe different scenarios where a dispute can arise and how it can be resolved. Note: We should meet with Roger to get his insights and similar experiences in Guifi.net. What follows will be a brief description of different scenarios where a conflict might arise and accordingly a description of different events that each of a such conflicts may trigger. The discussion will involve different actors/agents at different *levels* (as I have understood after my recent meeting with Leandro that there can be multi-tiered conflict resolution.) of conflict resolution [3] (page 7). The purpose here is to highlight the limitations in such an approach in terms of delay that it causes and the overall trust issues related to the discretion of the involved parties. The ultimate purpose of this section is to provide the motivation for a *self-contained, trusted, and distributed system* capable of both record-keeping and conflict resolution and then later we will assert that blockchain-based solution can be such a contender.

1.4 Conflict scenarios

Things to take into consideration. See [3] (pages 7 and 17)

1. Multi-tiered scenarios/conflict resolution.
2. Agents/actors involved in each such tier and their roles/authority.
3. Commons model and the need for compensation system. Ref: Wikipedia's def of commons model.
4. How different tiers are triggered? (Initial State)
5. How a decision is reached and verdict enforced? (set of possible results)
6. Ramification of a verdict (can there be further contention)?
7. Does the system ultimately reach consensus?
8. Compensation types (in terms of micro- and relatively macro payments (for motivation, VAT dynamics can be explained briefly for motivation)).

See the discussion about the compensation system in [1].

2 Blockchain-based solution

In light of the above discussion we will, hopefully in detail, provide motivation for a blockchain-based solution. The focus will be on its distributed and immutable record keeping with distributed consensus mechanism in place. Further we can talk about the provision of automation in terms of smart contracts. The main purpose of this section should ideally be to present a plausible case in the favour of *self-contained compensation system*. Ideally we should provide analogies with the above mentioned conflict scenarios and how they translate to blockchain's automation. We may have to talk about that further data (e.g., data that could be fetched from outside the chain) is usually needed while working on each tier of a conflict resolution. Then we can loop in the concept of Oracles and explain, in conceptual terms, how they work in unison with blockchain and how the original premise of immutability and trust is kept intact while doing so (maybe up to some extent?).

Following we can discuss a bit about different design choices available when it comes to design a blockchain-based system.

2.1 Permissioned vs permissionless blockchains

2.2 Oracles

Skim through [3].

2.3 Description of Swarm

Article to be described here: [4]

1. SWAP (Transaction): Swarm accounting protocol
2. SWEAR (Commitment): Registered nodes and Ensured ARchival
3. SWINDLE (Enforcement): Litigation on loss of content

3 Experimental setup

3.1 Network setup and architecture

1. Installed go implementation of Ethereum on Ubuntu 16.04 called geth[4].

TODO: Network architecture schematic. Ref: Figure 1 is still a work in progress. The details about setting up the local nodes and provisioning them with Ethereum nodes.

[3] `http://www.oraclize.it/`

[4] `https://github.com/ethereum/go-ethereum/wiki/Installing-Geth#install-on-ubuntu-via-ppas`

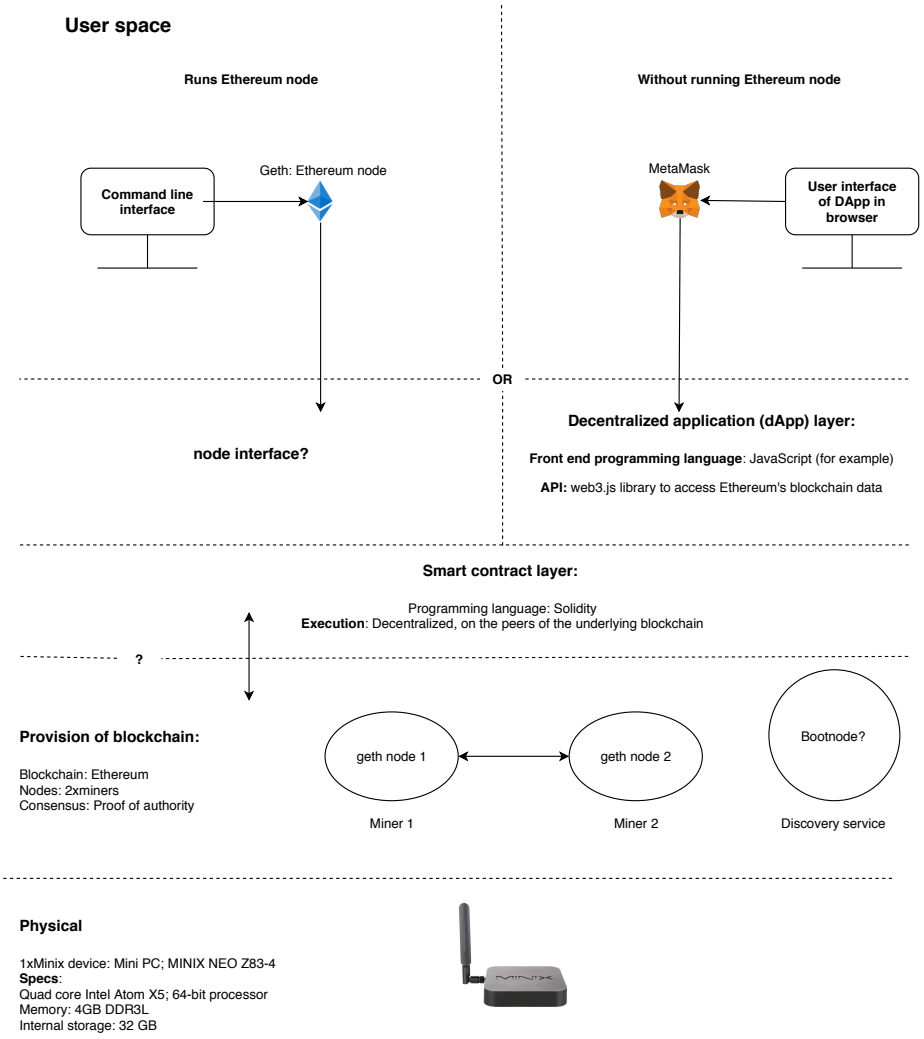

Figure 1: Overview of the experimental setup

3.1.1 Ethereum

A brief description of how Ethereum's blockchain works along with brief description of different terms such as gas, what Metamask is, what is a dAppp, concepts related to smart contracts (talk about ABI as well), and most importantly the description of *events*[5] and *call backs* that together realize the concept of oracle. Text before this will be removed/edited. Manos: Some clarifications

- Metamask: Metamask connects to a local chain (or local node of a public chain) through RPC or Websockets if you provide the corresponding IP and port. It also allows you to connect to chains without connecting to a local node through the INFURA service (only to the chains supported by them)
- Smart Contract Layer: Execution happens by the EVM (Ethereum Virtual Machines) that run inside each geth node.

Geth: Ethereum node:

Rinkeby and other Ethereum test networks:

Proof of authority: Proof of authority is an alternative to Proof-of-Work mainly used to setup private chains. A brief one-liner comparison with Proof-of-Stake with a plausible reference.[6],[7].

3.2 Campaing's set of smart contracts

A brief introduction of how Campaign's smart contract works and how it has been implemented on the local network setup and its extension [8].

3.3 Oracle's set of smart contracts

Cite this work for an argument about the trust surrounding the Orcales: [9] General description of all the main component smart contracts as in [10] and how these are updated according to our use case.

1. Dispatch
2. Lookup
3. API
4. Test smart contract

[5] http://solidity.readthedocs.io/en/v0.4.21/contracts.html#events
[6] https://hackernoon.com/setup-your-own-private-proof-of-authority-ethereum-network-with-geth-9a0a3750cda8
[7] https://wiki.parity.io/Proof-of-Authority-Chains
[8] https://github.com/StephenGrider/EthereumCasts/tree/master/kickstart
[9] https://www2.deloitte.com/content/dam/Deloitte/ie/Documents/Technology/IE_C_BlockchainandCyberPOV_0417.pdf
[10] https://github.com/axic/tinyoracle

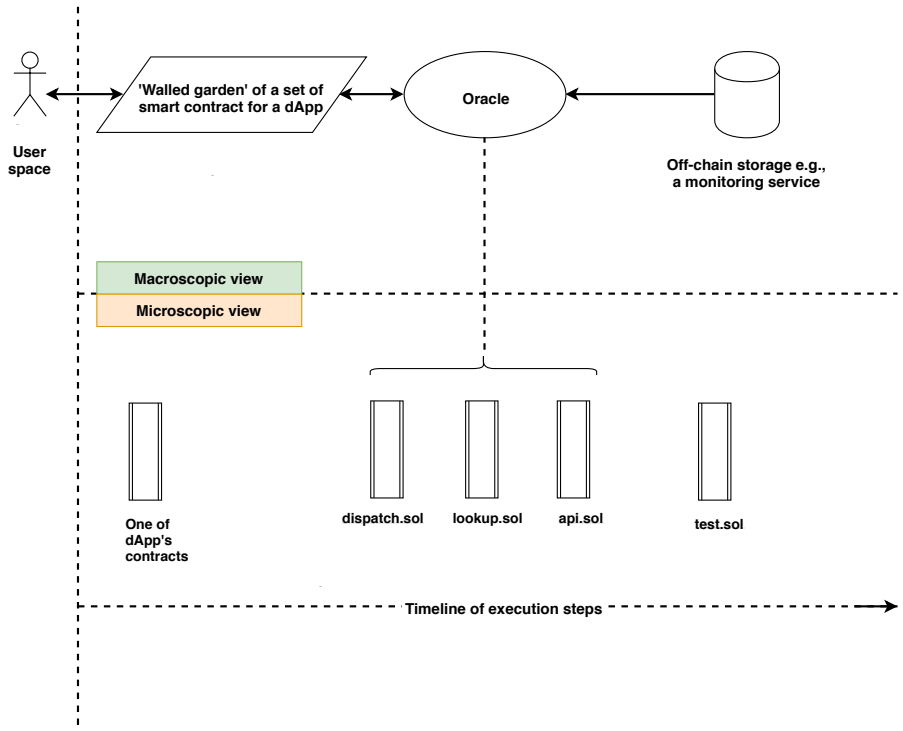

Figure 2: Understanding oracle

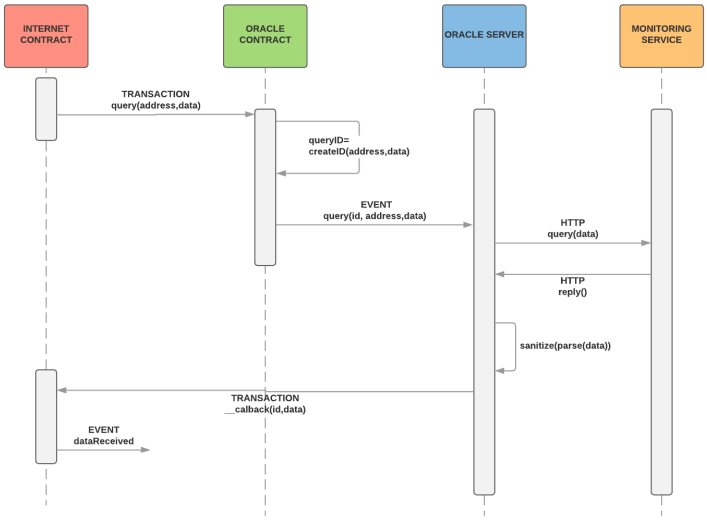

Figure 3: Execution of oracle. *Credits: Manos*

TODO: A schematic describing how these four smart contracts work in unison with each other. This will go along with the dummy db.json database.Figure 2 is still work in progress.

3.4 Internet Sharing's set of smart contract

Mainly the current state of the Internet Sharing smart contract. Will talk about the extent to which it tries to implement an economic/compensation model. Assumptions made. Ideal goals and objectives. Challenges, in particular to when it comes to the extent to which one can actually model, inclusive of all the conflict resolution scenarios, an economic sustainability model.

4 Discussion and future directions

4.1 Trust analysis with oracle

Here we could probably describe in detail how our outside data (i.e., external to blockchain) might look like and where it will be hosted. We could also talk about the level of trust on this outside hosted data. Describe relevant material in *The Blockchain as a Software Connector* [5].

4.2 System compliance

The purpose of this subsection will be to analyse how far we will be able to make our system self contained in term of making it in compliance with the policies and laws pertaining to the compensation use case under consideration. The issue of the extent to which this compliance might go is a critical and important issue as it will determine at what level we might need a human intervention and will we be able to back this intervention with the blockchain's underlying data storage and automation, if so, then again, upto what extent [6]? We could also talk about different litigation ramifications and how our system might be used to provide evidence as per a specific conflict.

4.3 Moving ahead: A road to DAO?

Ideally, here we could perhaps talk about the end product/prototype we are envisioning to deliver. We could also talk about its ultimate goals and purposes. And maybe we can finish with a description of *decentralized autonomous organization (DAO)* as our (or part of our) end product. Read the documentation at [11], [12]

[11] `https://www.ethereum.org/dao`
[12] `https://github.com/ethereum/wiki/wiki/White-Paper#decentralized-autonomous-organizations`

4.4 Ammbr's perspective

Mainyl a brief case in favour of how the discussion in this report would be able to benefit Ammbr.

4.5 Compensation system and data provenance

The purpose of this section will be to highlight the similarities, particularly when it comes to the system being in compliance with the relevant policies, between the current use case and my PhD's data provenance research. My focus will particularly be on the lessons learnt and insights gained while developing a blockchain-based solution for the compensation system self-contained and in compliance with the local policies and laws.

5 Acknowledgements

¡Place holder¿

References

[1] Roger Baig, Lluís Dalmau, Ramon Roca, Leandro Navarro, Felix Freitag, and Arjuna Sathiaseelan. Making community networks economically sustainable, the guifi. net experience. In *Proceedings of the 2016 workshop on Global Access to the Internet for All*, pages 31–36. ACM, 2016.

[2] Mennan Selimi, Aniruddh Rao Kabbinale, Anwaar Ali, Leandro Navarro, and Arjuna Sathiaseelan. Towards blockchain-enabled wireless mesh networks. *arXiv preprint arXiv:1804.00561*, 2018.

[3] Roger Baig, Ramon Roca, Felix Freitag, and Leandro Navarro. Guifi.net, a crowdsourced network infrastructure held in common. *Computer Networks*, 90:150–165, 2015.

[4] Daniel A Nagy Viktor Trón, Aron Fischer and Zsolt Felföldi. swap, swear and swindle incentive system for swarm. `http://swarm-gateways.net/bzz:/theswarm.eth/ethersphere/orange-papers/1/sw%5E3.pdf`, 2016. [Online; accessed 18-June-2018].

[5] Xiwei Xu, Cesare Pautasso, Liming Zhu, Vincent Gramoli, Alexander Ponomarev, An Binh Tran, and Shiping Chen. The blockchain as a software connector. In *2016 13th Working IEEE/IFIP Conference on Software Architecture (WICSA)*, pages 182–191. IEEE, 2016.

[6] Karen Yeung. Regulation by blockchain: The emerging battle for supremacy between the code of law and code as law. *SSRN*, 2018.