Fully Distributed GRIDNET Protocol, with No Trusted Authorities

Rafal Skowronski
Department of Computer Science
Poznan University of Technology, Poland
rafal.skowronski@cs.put.poznan.pl

Abstract— **The flow of information among people in today's world is essential. People need to exchange data, but they also need to store larger chunks of data for future retrieval. Various business schemes have grown by feeding themselves on these assumptions. Some of them provide the needed infrastructure, such as, cables or wireless base stations in case of GSM/LTE networks, while others provide complementary storage capabilities (cloud storage services). In this paper, we introduce a *Fully Distributed GRIDNET* protocol (*FD-GRIDNET*). It facilitates a solution to a problem of motivating users to intercede in a data exchange. MANET/DTN networks were envisioned as a target environment, however we do not restrain our protocol by design only to such. *FD-GRIDNET* is the first fully distributed data exchange protocol, which rewards intermediaries with a cryptocurrency, one created on behalf of the described communication system itself. It constitutes a communication system with a closed economy cycle, where acting as a router earns cryptocurrency, which in turn can be used for one's own needs, such as, but not limited to – data transmission. Indeed, *FD-GRIDNET* can be said to facilitate a cryptocurrency of its own. It builds upon a proof-of-work concept, but introduces elements of proof-of-stake as well.**

Keywords— mobile networks; Ad-hoc networks; Disruption tolerant networks,consensus, cryptocurrency, P2P

I. INTRODUCTION

US Air Force established a wide area network in the early 60s, as a result of, seeking a system, which would survive a nuclear attack [9]. In the late 80s commercial communication technology, had begun turning away from circuit switched networks towards, a more efficient, packet switched networking. Somewhere along the line a TCP/IP protocol stack was born laying a foundation to the INTER-NETwork, world wide data packet exchange environment, we are so familiar with, today.

Global data surveillance systems are no longer a subject of theoretical speculations. These are a matter of facts (PRISM) [10,11]. It is a common believe that one should have a choice with whom he or she wants to share her private communication with, - be it the very fact of communication alone (*metadata*).

Digital communication among people has come to rely almost exclusively on telecommunication companies. Whilst the system works well for most communication needs and is widespread enough in urban areas, it still suffers from inherent weaknesses of the trust based model. In many countries, telecom companies are forced by governments to record whole communication episodes or at least to facilitate storage of communication events - the so-called metadata. What is more, many rural areas in the world lack required infrastructure and so, they are not able to easily communicate at large distances at all. Various protocols and projects, including open-source, have been developed with the purpose of aiding low-cost communication among people; incorporating various types of MANET-style, or pocket-switched variations of networking, coupled by hardware, such as, Wi-Fi interfaces. Most, if not all previous approaches, however, lack a proper handling of privacy, security and what is also of key importance in our believe - they do lack a satisfactory and universal rewarding system for intermediaries. There have been multiple proposals to this dilemma, undertaking various angles. Some of which are based on reputation metrics, others appraise economic solutions; none of them however incorporate universal token, a wealth, which could be used beyond the needs of data exchange if such a need arises. What is more, none of them appraise a consensus mechanism which would allow for operation without reliance on a trusted-third-party.

We believe, that relay nodes are a backbone of every distributed packet-switched type of networking and if they are to participate on their own will, providing a proper incentive to co-operate is of crucial importance. In game-theoretic terms, cooperation in mobile ad-hoc networks, but also in any other type of networking - where relay nodes are considered to be autonomous while not having clearly defined benefits from participation in data dissemination; poses a dilemma. Nodes may be managed by different authorities, having different priorities. To save battery, bandwidth and processing power, nodes should not forward packets for others. If, however a significant number of nodes adopts such strategy, quality of network degrades vastly for all.

What is needed, is a fully distributed communication system, which rewards intermediaries based on the amount of

data which they help to disseminate. This communication system shall lack any kind of trusted third parties while being robust and immune to various kinds of impersonation attacks and hideous behavior.

Let us introduce *FD-GRIDNET protocol* - the first fully distributed communication system, to incorporate distributed ledger database, also termed colloquially as a '*blockchain'*. We utilize the concept *of proof-of work*, as well as, *proof-of-stake*. *Proof-of-work* mechanism has already been used by various cryptocurrencies and popularized initially by *Bitcoin* [6], which in turn, paved the path for others. Bitcoin used the *proof-of-work* concept to reach consensus on the state of a distributed ledger database containing transactions between users. When considering a communication system which is to provide a fair spread of rewards for the intermediaries, we inevitably need to keep track of packet deliveries. In *FD-GRIDNET* every data flow is being tracked and resolved through a blockchain. No communicated data is ever stored anywhere and communication identifiers are completely anonymous. Specialized nodes verify communication flows and *relay tickets*. The byproduct of the process, being a virtual currency, spread among relay nodes in accordance to their contributions. In our design, *proof-of-stake* is used to minimize hideous incentives, as well to allow for a higher network throughput, when compared to pure proof-of work mechanism, such as, the one used in Bitcoin. Each and every user needs to perform a certain *proof-of-work* related to one's identity before being able to participate inside of the network. This proof would then serve as a proof of stake. When a node begins to cheat, it will most likely lose its difficult to achieve stake.

Summarizing our contribution in this paper, we propose a first fully distributed communication system, which attempts to resolve the problem of rewarding intermediaries for their work and resources. We also tackle the problem of handling misbehaving nodes and upraise a solution based on voting and a *proof-of-stake*. In literature, there have been various proposals on how to handle misbehaving nodes. Monitoring and reputation schemes come at a price. They require overhearing of transmissions from others. Please refer to a work by Sonja and Jean [1], for a further discussion and review of previously known solutions. Due to a high level of generalization, our protocol applies to a wide range of usage scenarios; from cloud-based solutions, where nodes are rewarded for providing their storage capabilities, to *peer-to-peer* file exchange systems. Most interestingly, our protocol allows for creation of an anonymous communication network, in which, colloquially speaking, *it pays to participate*. *FD-GRIDNET* is a suitable choice for deployment in a mobile wireless scenario, as well as, for use on top of the Internet.

Marti, Giuli, Lai and Baker [7] consider the problem of relay nodes non-forwarding. There have been various other proposals, many of which present sound optimistic images, such as, in case when relaying node do not overhear the other one retransmit a packet within a given timeframe - the sender is notified about a faulty node. These proposals however do not envision at all what happens when many nodes collude to take down legit nodes with the aim of performing a Denial-Of-Service attack. There has also been no incentive, so far, to protect against a case where a blacklisted node generates a new identity just after being excluded. Previous works overlook to mention where such a distributed knowledge shall be maintained. If, we store reputation-related information at each node, how do we make sure that the information stays in synch? And how do we know that an attacker is not injecting packets with false information? Previous works seem to focus solely on selected aspects of the picture and, thus, turn out to be largely impractical and theoretic. SORI [2] is one of the selfish-behavior detection algorithms which considers security and utilizes a hash-chain for dissemination of reputation data. This however is effective only when nodes are not in motion and reputation data can be calculated upon a given area of constant, motionless members. Due to a lack of a global source of information, new nodes that come into vicinity remain unknown. It is quite visible that, one of the main troubles lies in the lack of protection against spawning of a new identity right after being exiled from participation.

The idea of economic or pricing-based schemes for rewarding intermediaries is not new - [3], [4], [5]. However, these proposals either require a temper proof hardware [2] or trusted authorities [4], [5]. In our solution, the system is guaranteed to provide a spread of goods in accordance to the intended ruleset, provided that honest nodes collectively control more CPU power than any cooperating group of attacker nodes.

II. Formulation of a Problem

We formulate the problem as follows; description of constraints shall follow along. We want node A to be able to send data to node B through an arbitrary number of intermediaries. Only node B shall be able to read encrypted data. No single intermediary nor data receiver shall be able to get to know the path traversed while data in transit so as to improve protection against statistical analysis etc. Intermediaries shall not be able to cheat the incentive system, for example, by adding an arbitrary number of intermediaries to the datagram for the purpose of increasing profit. It should be unworthy, in game-theoretic terms, for a current given intermediary to be willing to create additional artificial intermediaries, so as, to maximize profit. Any given intermediary shall not be able to remove any previous intermediary from the list of nodes traversed, so as to steal or maximize profit. The system shall payout credits only after successful delivery of data.

A solution to this problem should work under following additional constraints:

- Each node cannot assume existence of symmetric connectivity between any other node. Data between inter-mediaries should be able to be exchanged in a pass and forget manner when opportunity arises

- Nodes should be operational and able to make valid routing decisions (whether to retransmit or not to, whether the transit has been paid for or not) even without access to the entire blockchain
- Each intermediary should have no certainty of who the recipient actually is and whether he has just handed the datagram to an actual recipient or not.
- We do not want to enable any data exchange without ensuring data originator's balance can cover the costs. A single data transmission, though, usually consists of a very large number of packets. Verifying each packet against originator's balance would create unbearable amount of network overhead and delays. In a most optimistic scenario, a single verification per data stream should suffice.
- The recipient of the data should not be able to collude with data originator to maintain free communication
- Perfect forward secrecy shall be maintained at any given point.

To our best knowledge there is no known protocol which would come close to fulfilling above described constraints.

As in the ADON protocol, here notion of packet is closer to a self-contained peace of information and so, the term datagram will be used from now on.

III. Proof-of-Work and Proof-of-Stake

U.S Federal Reserve notes have not been redeemable in gold since January 30, 1934 [8]. For an average citizen, the value of currency lies in its limited supply. Its value is artificial but limited and should be determined by forces of supply and demand.

In our protocol the incentive behind data delivery is cryptocurrency, which's limited supply in turn is governed by the laws of physics. One needs to consume time and energy, in the form of electricity, to come up with an appropriate *Proof-of-Work* for a given transaction block. When one earns cryptocurrency, he can *consume* it for the purpose of generating a Transmission Token (*TT*), - which will be described in following sections. *TT* allows intermediaries to verify sender's willingness to cover data dissemination fees. *TT* can be thought of as a financial bond without holders specified until the data is delivered. Every intermediary however can verify bond's authenticity and hope to receive its fraction by cooperating.

Secondly, in game-theoretic terms one should not risk more than the expected return from investment. That is where the *Proof-of-Stake* comes into play. Every node needs to compute a *proof-of-work* of their identifier. This *PoW* consists of a hash value which's numerical representation needs to be under a certain threshold defined as *work difficulty*. It is coupled by a *nonce* - an integer value which results in a hash of the address to be under a given threshold. In case of a nonce overflow, the address is concatenated with itself until success. The result serves as a *Proof-Of-Stake*. It is stored in the blockchain together with one's address and is also attached to every datagram generated by a given address. In case of a lack of payouts from the data originator or due to its misbehavior an unfair node can lose its stake – he would lose the initial proof-of-work attached to his *address*.

IV. The notion of Transmission Tokens and Transmission Token Pools

In a shortest sense, a token pool is a data structure which is represented by a hash chain. A single token pool is defined by 100bits of a SHA256 hash-seed value, the number of hashes present in a hash chain and a final ceiling value of a given hash-chain.

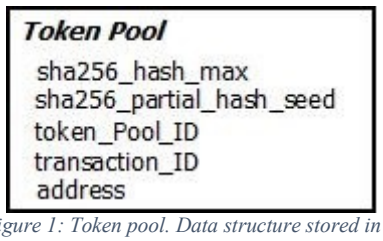

Figure 1: Token pool. Data structure stored in a blockchain

There are also fields binding a given hash-pool to its owner's address and a reference to a transaction in which coins were consumed for the purpose of generating a *Token Pool*. *Hash pool* is computed by a person sacrificing coins, and so, the person is the only one with knowledge of hash values between the partial-hash-seed and the final ceiling hash. Initial seed-hash, in its entirety, remains a secret until depleted. The number of seed's known bits is enough to prevent collisions (nodes detect end of a token by these bits) and secretive enough to thwart feasible brute forcing. Final hash can and needs to be known to the public. Coins sacrificed to generate a pool are not ultimately lost however. Every single hash, or a range of hashes from the parent *token pool* can be used to create a single *transmission token*.

Transmission Token (TT) is a data structure, by which, data originator authorizes transmission of a datagram.

Figure 2:Transmission Token, signed by sender

A *token pool* was generated by consuming[1] a certain amount of currency and now each hash in a token pool represents a share in a consumed amount of currency. In other words, by sending currency to an unredeemable address, one exchanges

[1] Currency is not consumed i.e. destroyed but deposited for future intermediaries.

coins for hashes in a hash pool. Value of a single hash in a given hash-pool can be calculated as $\frac{NumberOfConsumedCoins}{NumberOfHashesInAHashPool}$.

Single datagram is coupled with a single *Transmission Token*. It specifies a *transmission reward (TR)*. A single *TT* can reference a single hash from a token pool or a range of hashes by specifying the number of hashes being revealed. This way, data originator can proportion priority of a datagram. The higher the *TR*, the higher incentive for intermediaries to store and forward information for longer periods of time. On the other hand, the diminishing profit for further intermediaries, as a side effect, prevents network from being flooded by old irrelevant datagrams. This facilitates a *Time-To-Live* mechanism, one guarded by forces of supply and demand.

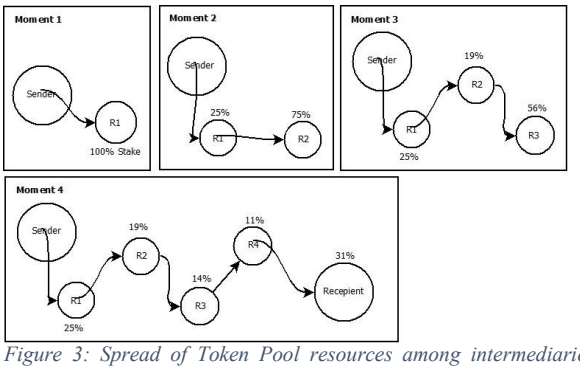

Figure 3: Spread of Token Pool resources among intermediaries via a Transmission Token

Figure 3 shows how transmission reward is spread among intermediaries. When a datagram reaches first intermediary, the first intermediary possesses a full share in a *TR*. When a current node decides to retransmit to another node, the current one hands over 75% of its current share to a next node. *The scheme repeats*. This way, it is unworthy for a current node to add artificial, owned by itself, payout addresses to a *hop-list,* since, it already owns everything it can. On the other hand, each node knows that, most probably, it needs to pass the data on, as coming into a direct contact with a target node might not be probable enough. Time is also of the essence. Data originator might be wealthy enough to have hired multiple initial messengers. Nodes constituting a given path, would receive credits, but only after a successful delivery. How the hop-list is protected against tempering among intermediaries will be described later on.

V. Storage of Information as Part of a Distributed Blockchain, Merkle Trees, Reaching Consensus

In our system, every crucial information is being stored as part of a distributed ledger. Each entry in a ledger is called a block. Blocks are bound together – each following block contains a hash value of a previous one. The more blocks, the harder it is for an attacker to replace a given block. The difficulty lies in a required computational power; This is mainly because data contained inside of a block needs to contain a hash which meets a certain *difficulty criterion* i.e. - it needs to be below a certain value. Full nodes – the ones which store the entire blockchain, rival among each other, to find a proper nonce – a single value inside of a block. Its target value results in block's hash to be below a difficulty cap. Block with a higher difficulty wins - it is accepted by other nodes and added to the blockchain. That is how consensus among different *mining nodes* is reached. Attacking the scheme would require an immense computation power. The more blocks, the more secure the blockchain.

There are full nodes – ones which contain entire blockchains and there are lite-nodes. Full nodes verify transactions; they need a high computational power. Lite nodes are fully functional nodes; they do not possess entire blockchain and ask for needed data when needed.

Single block consists of a *merkle-tree,* which in turn, contains dozens of transactions. When a lite-node wants to verify a transaction, it does not need to query for a whole blockchain, or even an entire *block*. What it needs, is to query a *full node* for a single *path* inside of a merkle-tree which contains the - to be verified - transaction.

Long story short - That is how many *Proof-of-Work* based crypto-currencies are implemented. Let's now focus on how a typical *proof-of-work* mechanism relates to our design and proposal. The main difference lies in how full nodes verify transactions. In our case, full nodes verify standard cryptocurrency transactions but they do verify *information flows,* as well. There is a need for full nodes to understand that currency is deposited for future intermediaries and that they might need to be able to redeem that deposit. This is also one of the main reasons why *FD-GRIDNET* cannot be implemented as a meta-currency[2]. In our design, information flows are designated by *Transmission Tokens*.

Each full node verifies whether a given hash from a particular hash pool had been already utilized. A situation might arise, when a hash from a deeper i.e. closer to the seed, part of a hash pool had been used already for another transmission. Each node keeps track of its current hash pool utilization state. Thankfully, storage of a hash-pool is memory efficient. Hash pool suitable for dispatching of 100,000,000 datagrams consists of a 256-bit hash-seed, 256-bit ending-hash and an integer. In case of hash re-use, a given node is blacklisted inside of a blockchain; preventing its future datagrams to be relayed by others.

VI. Structure of a Datagram and Data Transmission

In *FD-GRIDNET* there is no *PKI* infrastructure. Distributed blockchain serves as a global source of trusted information, with its integrity guarded by a proof-of-work mechanism. Users are identified by their addresses. Addresses

[2] Metacurrency is a cryptocurrency which does not facilitate a blockchain of its own, instead it builds upon an already established blockchain such as Bitcoin's.

are derived from user's public keys. Private keys are kept secret and shared with no one. Each datagram contains an anonymous sender's address.

Sample Identity Proof of Work

Sender address: 1BvBMSGYstWetqTFn5Au7m4GFg7xJaNVN
nonce: 123456789
number of concatenations:0
friendlyID: user1
proof-of-work:
000000000000006bc28cb4165ff97fb2b193b79e7496ad605662ea5f4b4d1b90

Figure 4: Sample identity-proof-of-work

In a scenario where there is no access to a full node i.e. to a blockchain, intermediaries are assured to some degree by a *proof-of-work* attached to sender's identity, such as one visible in Figure 4. Every *Transmission Token* is signed, so in case of *TT* reuse, or a non-existent *Token Pool* – sender would be banned, as soon as, an addressee initiates a clearance.

Initiating a payout is associated with a fee calculated as a fraction of a *transmission reward*. Therefore, a recipient might prefer to initiate clearance not too often for a single sender. This minimizes size of the blockchain and allows recipients and intermediaries to make profit. Each intermediary is free to check status of his payouts in the blockchain. In case of no payouts, in a reasonable time frame, one might decide to stop forwarding datagrams for a given user.

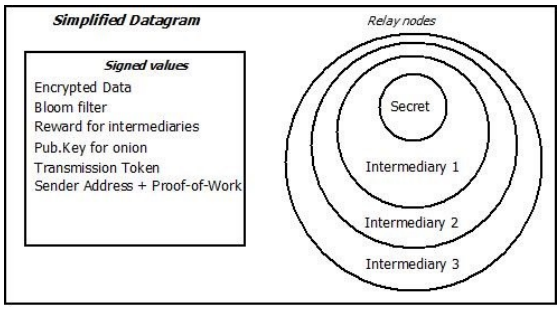

Figure 5: Simplified structure of a datagram.

FD-GRIDNET datagram consists of two key parts. First one is signed by a dispatcher. It contains a Bloom filter - used for routing decisions, a transmission token, sender's address - coupled by a hash meeting certain proof-of-work; it also specifies a reward pool for the go-betweens. Public key inside of the signed chunk of datagram is of special importance. It is used by intermediaries to add their addresses to a list of relay-nodes. Less a list, the data structure, reassembles an onion, with each layer encapsulating address of a single intermediary. Each relay node adds a new layer of encryption using the same, single public key made available by sender. There's no way for intermediaries, as well as, for the recipient, to get to know, or modify previous relay nodes. At the center of the onion there is a secret value generated by and known only to the sender. This thwarts attempts of separating list of relay nodes from the data, or a reuse of list for other transmissions. Single *onion* is valid only for a specific datagram. Upon clearance initiated by recipient, all the pending onions are delivered to full nodes, together with, their corresponding transmission tokens and secret values. These values are included into the blockchain. As soon as, a sender notices, transmission which needs to be approved inside of the blockchain, he releases, also, into the blockchain, the necessary private key required by full nodes to begin their work of clearing payouts.

VII. Discussion of security, privacy and friendly-id registration

Data segments inside of a datagram are encrypted with a symmetric cypher, with a passphrase encrypted using public-key cryptography. Public keys are stored inside of a blockchain. Sender needs to query its neighbors i.e. the blockchain to receive one's public key. Users can query for a PK by one's known friendlyID. FriendlyIDs were implemented as a kind of DNS system inside of a blockchain. It is worth to mention that due to a relatively immense amount of work required to generate a PoW attached to one's identity, various kinds of malicious behavior can be mitigated by banning one node from participation based on various heuristics through voting.

VIII. Reverse Situation

FD-GRIDNET also supports a reverse situation, one which was omitted in this paper for clarity. Suppose a node is looking for a specific information, a file for example. He can notify his neighbors about the fact and propose a bounty. Relay nodes, which deliver, would be rewarded in a way similar to hereby described.

IX. Future-work

Currently we are working on the implementation. We try to better formulate the incorporation of hybrid proof-of-work/proof-of-stake mechanism inside of the blockchain for increased data throughput.

X. Applications and Summary

To summarize, we have proposed a practical solution to a problem of rewarding intermediaries for their work based on a blockchain technology. Various technical details have been omitted in this paper for clarity. Applications of this protocol are vast. Ranging from incorporation inside of MANET/DTN networks, to P2P protocols and cloud based storage services.

References

[1] Self-Policing Mobile Ad-Hoc Networks by Reputation Systems, Sonja Buchegger, Jean-Yves Le Boude
[2] SORI: A Secure and Objective Reputation-based Incentive Scheme for Ad-hoc Networks
Qi He Dapeng Wu, Pradeep Khosla
[3] L. Buttyan and J. Hubaux, "Enforcing service availability in mobile ad-hoc WANs," IEEE/ACM Workshop on Mobile Ad Hoc Networking and Computing (MobiHoc), Boston, MA, USA, Aug. 2000.
[4] M. Jakobsson, J. Hubaux, and L. Buttyan, "A micro-payment scheme encouraging collaboration in multi-hop cellular networks," Proceedings of Financial Crypto 2003, Gosier, Guadeloupe, Jan. 2003.
[5] S. Zhong, J. Chen, and Y. Yang, "Sprite: a simple, cheat-proof, credit-

based system for mobile ad-hoc networks," IEEE INFOCOM 2003, San Francisco, CA, USA, April 2003.
[6] Bitcoin: A Peer-to-Peer Electronic Cash System, Satoshi Nakamoto
[7] Mitigating Routing Misbehavior in Mobile Ad Hoc NetworksSergio Marti, T.J. Giuli, Kevin Lai, and Mary Baker Department
[8] U.S Government, Gold Reserve Act , January 30th 1934
[9] History of Telecommunications Technology, Christopher H. Sterling, George Shiers
[10] Gellman, Barton; Poitras, Laura (June 6, 2013). "US Intelligence Mining Data from Nine U.S. Internet Companies in Broad Secret Program". The Washington Post. Retrieved June 15, 2013.
[11] Braun, Stephen; Flaherty, Anne; Gillum, Jack; Apuzzo, Matt (June 15, 2013). "Secret to PRISM Program: Even Bigger Data Seizures". Associated Press. Retrieved June 18, 2013