Alishan Hassan
December 7, 2015
Cryptocurrency Cabal

Encrypted, Anonymous, Network with Bandwidth Derived Cryptocurrency Rewards and Transactions

Formerly: A Bandwidth Based "Proof of Work"

Introduction

In this paper, I will attempt to describe a potential encrypted, anonymous, peer to peer, network that keeps track of bandwidth usage and rewards nodes for expending bandwidth necessary to keep the network running. The rewards are effectively a form of currency that can be used for transactions within the network by using the public address of each of the nodes in the network.

Motivation

In circles not familiar with cryptocurrencies, there is often resentment with the wasteful nature of Bitcoin's proof of work. People wonder, why should so many processing cycles, and therefore energy, be spent doing useless hashing? Why can't Bitcoin mining computations be used for something that's beneficial to humanity, such as protein folding or endeavors such as SETI@home? The reasoning is largely because the act of wasting energy is spent on making sure the Bitcoin network is secure and stable and provides a massive barrier of entry for any nefarious users willing to attack the network. The wasteful nature of the proof of work is intrinsic to the Bitcoin protocol. Running other computations compromises the integrity of the network.

Regardless, the motivation behind this project was to find some mathematical operation that could serve as an alternative proof of work. Some endeavors already attempt this, such as Gridcoin[1] which is linked to the Berkeley Open Infrastructure for Network Computing (BOINC). BOINC allows volunteers to contribute their computer's processing power to solve larger, usually research orientated, problems such as gene mapping. Gridcoin offers a cryptocurrency payment for those who mine by contributing to BOINC. So how does this work? Gridcoin still has a proof of work system that is run side by side with the BOINC computations. Anyone who runs just the proof of work system gets a small number of coins, and anyone who runs both the proof of work and BOINC receives significantly more coins. The problem inherent here is that security is weaker than in Bitcoin. It does not possess the same level of protection against double spend attacks that Bitcoin has. Furthermore, the idea behind Gridcoin is that they don't want to waste energy simply doing hashes, but the system, nevertheless, has to do a normal proof of work on the side, anyway. The result is an unreliable currency, wasted resources to keep the

[1] http://www.gridcoin.us/

unreliable currency running, and BOINC not receiving all the available computing power of the miners.

There have been other attempts at similar, more altruistic, alt-coins such as CureCoin, but they all suffer from the same drawbacks. My initial goal was to try and find some sort of function that could provide benefits similar to SHA-256. My thoughts revolved around atmospheric calculations as a basis, however the best ideas I could come up with would have the predictability benefit of SHA-256. Bitcoin miners, on average, add a new block to the blockchain every 10 minutes. This is very consistent and the difficulty can be adjusted to keep it this way. Any of my attempts would have had no predictability at all, which would be deeply problematic for a stable currency.

Thus, it was suggested that I investigate wasting other resources, besides energy. Storage was the first suggestion. In a way, one could be rewarded for opening up the extra storage on their hard drives for a cloud storage solution. This could be an excellent way to share files and store backups on the cloud, while rewarding people for providing storage (as opposed to paying a cloud storage service separately). There are a couple attempts at this already, such as PermaCoin[2] and storej. The former, however, acknowledges the importance of Bitcoin's proof of work and, instead, is supposed to be a modification of Bitcoin rather than alt-coin. It could potentially add decentralized storage to Bitcoin while maintaining Bitcoin's underlying incentives. Implementing a functionality on top of Bitcoin is certainly a way this project can go.

Ultimately, I chose to pursue the usage of bandwidth as opposed to energy. There are certainly some interesting things to consider about bandwidth. Some Internet Service Providers, such as Comcast, have begun to treat bandwidth like a limited resource. In some markets, users are limited to 300 GB of bandwidth per month in consumer applications. Thus, could bandwidth be used to back a currency? And, if it could be, how exactly would the bandwidth be expended by users?

Background

Let's consider an anonymous, encrypted, peer to peer, network. We can say it's similar to Tor (although Tor, technically, isn't strictly P2P) with encrypted data being sent from one node to a couple random relays before reaching an exit node that connects to a destination. The relays and the exit node are expending bandwidth to keep the Tor network running. If bandwidth is, indeed, valuable, then could we reward users who expend bandwidth with some sort of currency? After all, with internet caps and bandwidth limitations, why would someone expend it all on running a network? If they wanted to download a file anonymously, they could just use Tor without acting as a relay or use a torrent with a VPN to achieve some level of anonymity. It certainly seems plausible that a currency system could provide an incentive to keep an anonymous network running, so the next thing to do is to look into different types of anonymous networks.

The first network to look into is called Dissent[3]. Unlike Tor, which uses onion routing, Dissent implements a DC-net, or dining cryptographers network. This allows for cryptographically-provable anonymity. The dining cryptographers problem is an interesting one. It was proposed in the late 1980s and involves three cryptographers trying to prove to each other

[2] Miller, et. al. "Permacoin: Repurposing Bitcoin Work for Data Preservation"
[3] Wolinsky, et. al. "Dissent in Numbers: Making Strong Anonymity Scale"

whether or not the NSA paid their restaurant bill. In essence, the waiter informs them that someone had paid for the meal, and it could be one of the three cryptographers or the NSA. Quite aptly for the cryptography community, the three customers respect each other's right to make an anonymous payment. Nevertheless, they want to find out if the NSA paid, so they use a two stage protocol. In the first stage, every two cryptographers establish a shared one-bit secret, say by tossing a coin behind a menu so that only two cryptographers see the outcome in turn for each two cryptographers. Then, in the second stage, they publically announce their bit. If they did not pay for the meal, they would use the exclusive OR (XOR) of the two shared bits they hold with their two neighbors and if they did, it's the opposite. This brilliant protocol sets the basis for a DC-net, which establishes the core of dissent. There are some issues, though. Dissent, while able to handle large anonymity sets, is extremely slow. It's so slow that it is not possible to perform real time interaction with a server or another person. The true strength is in the anonymity. A cryptoanarchist would benefit greatly with the ability to publish blog posts anonymously, expressing his/her dissent. On the other hand, the authors propose that Dissent could be used to enhance Tor, namely the issue it has with entry guards.

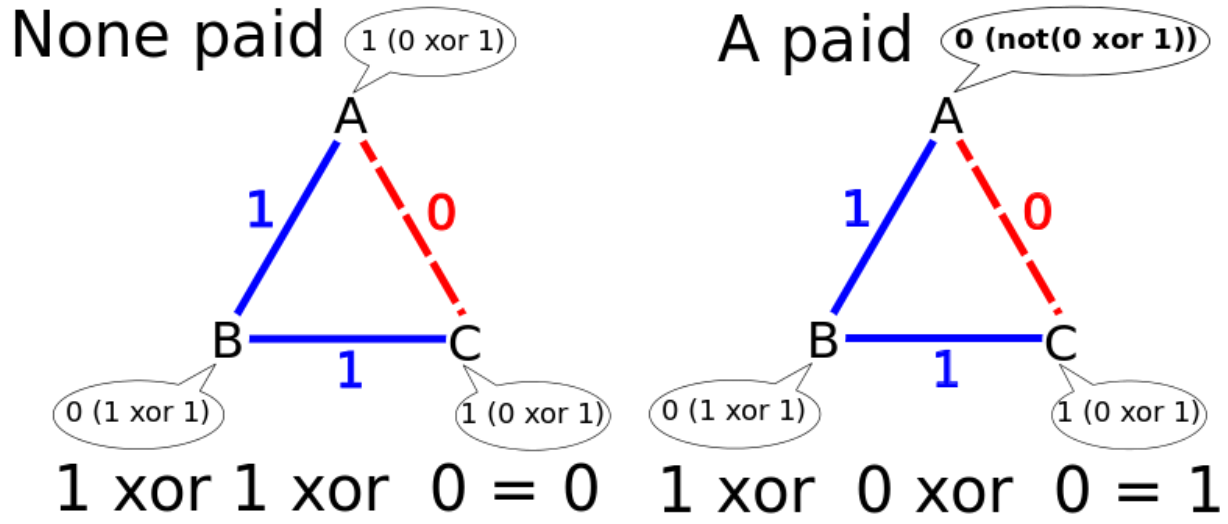

Figure 1: Dining Cryptographers
Source: https://commons.wikimedia.org/wiki/File:Dining_Cryptographers.svg

Another proposition was to use a type of circular network. I spent quite some time trying to find some implementation of a circular network, but nothing came up. The development of such a network is an interesting project in its own right. The idea behind this network is that every node is connected to each other in a circle. This means that one node is only directly connected to two other nodes. A basic depiction is given in Figure 2. Anonymous communication can only occur among the nodes of this network. Any sort of outside communication would have to rely on a single node, insecurely, obtaining data. The idea is that there is constantly a block of data that is being transmitted around the network. Using private keys, each node can check to see if a packet of data meant for them is available. If it is, the data is extracted and the node now has to re-encrypt the entire block of data and then send it to the next node in the network. This continually repeats. The problem with such a design is

tremendous latency. Data has to pass through every single node before it returns to the starting node, and if any node in between finds some data addressed to them, they have to go through the timely process of re-encryption. Such a network is not, at all, practical for real time usage, but definitely has some intrinsic benefits that are worth investigating.

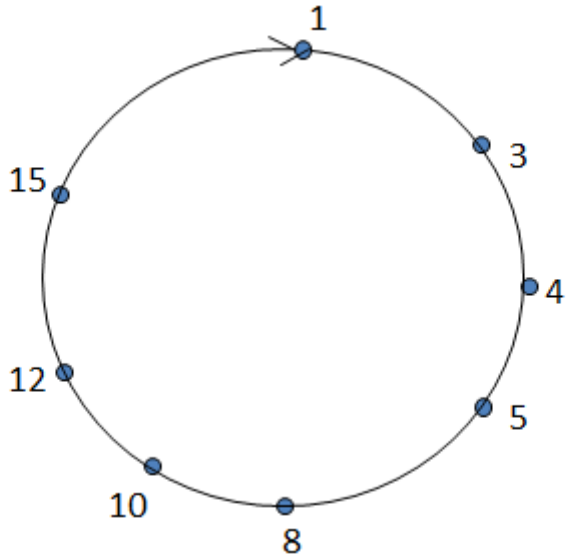

Figure 2: Circular Network
Source: http://blogs.cornell.edu/info2040/files/2012/11/Capture-2453y2u.png

Ultimately, the decision was made to simply expand upon the basic implementation of Tor. There is a level of simplicity offered that makes a proof of concept more usable. Implementing a currency on top of it does pose a few challenges, though. For example, how can we make the currency secure and stable? And how can we decentralize the system? Let's say you have three nodes literally next to each other and use them as relays. If you configure them to simply use each other as relays and not connect to anyone else, the owner could reap tremendous amounts of reward. So, how can we prevent this from happening while still staying decentralized? It's difficult, but perhaps we can implement some sort of limited amount of centralization.

Inspiration: Private Torrent Trackers

Prior to distributed hash tables, torrents were technically not centralized. They used trackers, which are servers distributed across the internet that tell users what torrents are available and learn what torrents, and their content, that have just been uploaded. The tracker knows what pieces each individual in swarm have and are able to direct uploads and downloads accordingly. There's also specialized trackers out there called private torrent trackers. These trackers require membership and a passkey, so that not anyone is able to join. They keep track of seed ratios as well as a user's upload/download ratio. In general, when the ratio is low, the user can get banned from the tracker, and lose the ability to download anything the tracker knows about. This keeps out leechers who are detrimental to the state of the network and provide nothing in return.

Can such an idea be implemented into this peer to peer network? Considering torrent trackers keep track of upload/download ratios, perhaps it is feasible to use trackers to track bandwidth usage. The bandwidth usage can be confirmed by other nodes by giving a log of the data transferred. The log would consist of entries that consist of an identification for the transfer, a hash of the previous transfer, a size (to quantify the bandwidth), a timestamp, and the addresses for each node involved plus the direction of the data transfer for each respective node. These trackers, by knowing the amount of bandwidth that is being used, can confirm bandwidth expenditures and reward nodes accordingly.

Overall Functionality

Now, again, there is an issue with one person acquiring a bunch of nodes and reaping rewards, or just lying about how much bandwidth was actually transmitted. How can this be solved? Well, let's consider how Tor works. In Tor, the data has to go through a couple relays before it reaches an exit node. As a first precaution, in a transmission if the nodes report different values for the bandwidth used, then something has certainly gone awry. So, let's say you upload a cryptoanarchocapitalist blog post, and one of the nodes reports a different value for the bandwidth than other nodes. Well your upload fails, you get a response telling you that, and you are required to try again. However, when trying again, you will get another set of randomly chosen relays, so that the previous relays don't cheat again. Your client software can be configured to keep trying until all of the nodes agree on bandwidth usage and the hash. Only at that point is a reward given out.

When it comes to selecting the relays, the network does so completely randomly with three constraints. The first is that it doesn't permit that a node within the same /16 subnet is selected, like Tor. Second, it makes sure that the ping for the nearest relay does not exceed 1000 ms (unless there is nothing else available) in order to prevent too much latency in the network. The third constraint is that if rejects a set of relays and can't find any others, it will refuse to function. In regards to the number of relays selected, there would need to be some computation of what offers the best balance of integrity and performance, however in this case we're sticking to the Tor standard.

Entry points (ie. the node of a user sending or requesting data) encrypt their data and make a hash of size of the data to send to the tracker so that it can doubly verify that the relays are reporting the correct bandwidth. Entry points do not earn rewards for bandwidth, however, since the network is working for their benefit. However, any entry point can be relay or an exit node as well. In fact, this would be necessary in order to prevent leechers from bogging down the network.

So, what about these trackers? Well, these are servers run by individuals. Their identities and locations are known and they keep a bandwidth log. This log is appended to a universal log which is similar to the blockchain. In a way, the bandwidth corresponds 1:1 with the reward and the transactions on the network occur as messages sent between public addresses. There is another value stored in the log that is constantly updated, a total transaction amount. What this effectively does is provide a positive or a negative number that is added to the bandwidth consumption. This determines, effectively, your current balance. While every individual tracker is incapable of monitoring the entire network on its own, the logs of the whole network are distributed across several trackers. Like Permacoin, if one goes down, the rest can compile and

backup the data that went missing. These trackers also have the ability to blacklist problematic nodes.

Future Plans

If anything, this endeavor does show that there is still a lot more to learn and lot more to test in regards to cryptocurrency proof of works, currencies backed by bytes (a byte standard, if you will), anonymous networks, and cryptography in general. There are two things I would like to test out. The first is devising a proof of concept of this anonymous network with bandwidth backed currency. We can already see that there are number of security holes, and that it doesn't come anywhere near Bitcoin in terms of providing a viable currency. Rather, it is a way to reward people to keep an anonymous network alive. As a technical exercise, I do believe it is worth pursuing.

Second, it would certainly be interesting to try and implement a circular network. In my research, I learned that many things I came up with have been tried and that others have thought up of some extremely elaborate and creative ways to implement cryptography. However, I was surprised at the complete lack of work on an anonymous circular network. Certainly there are some fundamental issues with nodes joining the network, security issues when communicating with some point outside the network, and the slow performance. However, as a system that could implement cryptocurrency payments, it seems very promising. It can be envisioned as a small, tight-knit, community that values privacy above all, communicates secretly, and does business only with members.

Final Thoughts

There is certainly a case to be made for adding some sort of reward system for expending bandwidth. But, there is also little doubt that such an implementation has to have a proof of work system similar to Bitcoin in order to have any real level of security and success. When coming up with my proposal, my approach the security flaws was to apply bandages, and that is not acceptable for a globally significant currency. If we did implement a Bitcoin-esque proof of work system, however, we would be doing the same thing Gridcoin does. It would appear that the reward is for some noble purpose (providing bandwidth for an anonymous network in our case, providing computing power to BOINC in Gridcoin's case), however the actual value would be coming from the robustness of the protocol made possible by the demanding, and wasteful, proof of work. Perhaps the real solution is to pursue a similar goal to Permacoin, which involves building upon the, already established, Bitcoin protocol and adding some sort of method to allow for anonymous communication. It certainly seems the protocol will strengthen anonymous networks, not the other way around. After all, as of now, there is no protection against double spending here.

There are also concerns about centralization. David Chaum, in 1990, had a centralized cryptocurrency called Digicash. It never caught on like Bitcoin and his company went under in 1998. Arguably, the reason it went under was because it didn't offer decentralization the way Bitcoin did, it was a central bank. There were very few competitors in the 1990s, so a partially centralized system does not seem promising today.

Additionally, there is also one big issue that really does need to be accounted for. With Bitcoin, energy is being spent on nothing but to maintain the stability of the currency. That energy has no other value (except, perhaps, as a space heater). With bandwidth, on the other hand, there is one tricky problem. The use of bandwidth is usually to obtain something of value, be it articles or cat pictures. And this kind of brings me back to an earlier question, can bandwidth fundamentally back a currency? If the value of bandwidth is the content of the bandwidth itself, then does adding a currency on top of it take away from the value of the content? Or, does it mean that the currency itself doesn't have any intrinsic value? Is this type of currency just a badge to show off, like points in a video game? Hard to say for sure, that would be up to the market to decide.