# Zerocoin, the really anonymous cryptocurrency

Anyone who read the newspapers of the last months has heard of Bitcoins, the electronic cash system: recent problems[1] have put the fast-spreading cryptocurrency in turmoil. Before this, huge speculation have massively increased its value, making early "miners" shares worth a fortune; now, the strength and frequency of value fluctuations is making everyone related to this economic world hold their breath. In addition to this, the elegance of the system, the fact that everyone can create its own money by investing time, and numerous other advantages[2], have made Bitcoin famous around the globe.

However, even as widespread as it is, the Bitcoin system isn't nearly perfect: some long-term privacy issues still remain. Moreover, as public opinion got shaken with the leak of classified information done by Edward Snowden, the revelations about PRISM, and other mass-surveillance programs, the need for personal privacy drastically increased. Not surprisingly, several solutions were proposed to palliate Bitcoin weak privacy protection: one of them, Zerocoin, a recent cryptocurrency, provides interesting functionalities to protect Bitcoin user's personal sphere.

This paper contains two main parts: first, a presentation about the Bitcoin system, its mechanics, and its problems regarding privacy; then, a presentation of Zerocoin, which is a solution to most of these privacy concerns.

## On Bitcoins

The concept of e-money isn't new at all: the first milestone was laid in 1990 by David Chaum, with his paper on anonymous electronic cash[3]. After that, Visa and MasterCard have made e-money very real. However, Bitcoin, created in 2009 by Satoshi Nakamoto[4], is different from previous solutions in several ways: it is completely decentralized; it doesn't have a central bank that you would have to trust. Bitcoin network is a peer-to-peer network, meaning that it doesn't have a single point of failure. It is intrinsically worldwide, has very low fees thanks to the absence of a central bank, and you can start receiving Bitcoins for free. More interestingly maybe, you can create your own Bitcoin, by running a special program on your computer: a *miner*; you don't spend anything else than time and electricity.

Bitcoin's main data structure is the *Block Chain*, a distributed append-only linked-list that can be seen as a global ledger, upon which every computer of the network agrees[5]. The Block Chain is made of *Blocks* themselves containing *Transactions*. Blocks are linked together by the hash of the previous block, forming the chain. Blocks can be seen as pages in the global ledger, and Transactions as individual records on the page. A Transaction has one or more inputs, and one or more outputs; the meaning is "collect the money from all these input, and send it, in these proportions, to the outputs"; it synthetize the action of sending money.

[1] Bankruptcy of MtGox, one of the main trading places for Bitcoins.
[2] To quote a few: Irrevocability of transactions, which is safer for payee; Resistance to Denial of Service attacks.
[3] [Untraceable electronic cash, D. Chaum]
[4] It is a pseudonym; the author's real name is still unknown.
[5] All computers of the network, at a given time, have the same view of the ledger.

Interestingly, Bitcoin doesn't use the very common notion of account/wallet to store your money: the very term "storing" isn't even appropriate. Instead, your own several past transactions (on which you are quoted in the outputs), effectively owning the value they represent; that way, strictly speaking, Bitcoin has no accounts. Yet, it is still a useful abstraction for a user's set of transactions, and might therefore be used with this meaning afterwards.

The chain is growing with new transactions: Miners are creating new blocks, which contains the transactions that users from the network want to perform and broadcast[6], validating them (by adding a *proof of work*), and broadcasting them to the network; if the block is accepted by the other miners (it is not creating a *fork* in the chain with some concurrent other block), the miner that validated it also receives a reward, in Bitcoins, for supporting the system: Hence the denomination "mining Bitcoins", process in which you are in fact validating others' transactions, and getting rewarded.

Forks happen when two miners try to add a different bloc to the chain. As this is not allowed, only one will stay as the next block, the other one being destroyed, its transactions re-broadcasted to be later included other blocks. Decision on which block is legitimate and which is not is made by comparing the proof-of-work, the validation added by the miner. These proofs-of-work are computationally hard problems that require a lot of calculations to be done, even by networks of computer working together. *Hashcash*, the proof-of-work used in Bitcoin, is a *publicly auditable*, *trapdoor-free*, *dynamically-throttled* scheme: everyone can easily check if the work was done, no "bypass" exists on the function which means everyone *has* to do all the calculations to produce an answer[7], and last but not least, the hardness, and therefore the required computation time, can be dynamically adjusted by the network. In Bitcoin, this is used to ensure that the creation of a block takes 10 minutes on average.

You may have noticed that up to now, we weren't speaking at all about encryption. Indeed, *everything* that was presented is public: the block chain, the blocks, the transactions. This implies, as transactions replace accounts, that "accounts", assets and money transfers are public.

Notice how it is different to say "the NSA can spy on the bank and discover how much I have" and "my neighbor, using only Internet Explorer, can check my account and what I'm buying": in Bitcoin, the latter applies.

Bitcoin's way to mitigate this undesirable behavior is using *pseudonymity*[8]: users' identities are not usernames, emails, or social security numbers, but *public keys*; they are unique, and generated by the Bitcoin client; given only the public key, the general belief/hope is that it is hard to identify the person owning it; in fact, it is desirable to have *several* public keys per user, to make this task even harder.

When trading with Bob, Alice has to share her public key, giving at the same time Bob the ongoing possibility to check how much money is attached to that key, what transactions are made. But if Alice has a lot of such keys, she can split her fortune on these different keys, giving Bob only a partial view of her activity.

[6] Send to every node in the network

[7] Some other scheme willingly have trapdoors; in spam mitigation, it is interesting that people that legitimately need to send lots of emails (to a mailing list) have a way to skip the computation and still produce a valid answer.

[8] While anonymous here means "without giving any identification information", pseudonymous means that you have to share a way of identification, but it can be fictitious; you can hide behind a pseudonym.

The pseudonymity of Bitcoin lies therefore on the fact that it is (believed to be) hard to map a public key with someone, or to link together public keys of the same individual (which would give an attacker a complete view of the activity).

Several analysis[9] have shown that the current solution is not satisfactory: even without investing too much computational effort in it, you can use attacks based on graph-theory, de-anonymization techniques[10] successfully experimented on anonymized social network data, to correlate someone's keys and online identities; along with this completely passive attack, active attacks, with participating nodes and traced Bitcoins, allows an attacker to recover even more information on the network, as well as on individual nodes.

In conclusion, regarding "anonymity", Bitcoin scheme is not sufficient.

## On ZeroCoins

Working in pair with Bitcoin rather than being "Bitcoin 2.0", ZeroCoin adds the privacy-friendly feature that was lacking in Bitcoin. ZeroCoin is a separate cryptocurrency, proposed in 2013. When this presentation was written[11] , ZeroCoins were still conceptual. ZeroCoin's strengths are based on the decentralized, trustless design, very close to the spirit of Bitcoin (notice how privacy from your neighbor is easier with a trusted central bank).

ZeroCoin's concept is to do a cryptographically secure currency swap, to allow users to anonymously trade Bitcoins, i.e. make non-public transactions. Non-public transactions, in addition to the obvious use of it (discreetly sending money to a peer), could be used to anonymize assets, by sending money to oneself to another fresh public key. ZeroCoin uses two main primitives for this: *Mint*, which creates a ZeroCoin, and spends the correct amount of Bitcoins, and *Spend*, which trade the ZeroCoin for Bitcoins. Once in the Zerocoin world, the amount minted becomes untraceable.

A useful analogy can be done with a public, shared corkboard, and a trusted guardian[12]. Suppose that you have 1$ on you, and you want to anonymize it in Bitcoin terms, meaning you want to change the owner, without the transaction being public. First, pin your 1$ bill on the corkboard, and ask the guardian for a voucher that states "Certificate: 1$ pinned". This voucher is anonymous; give it to a friend (conceptually, it can be you under a different identity), and ask him to get the 1$ back. When shown the voucher, the guardian will unpin *any* 1$ bill from the corkboard and give it to your friend. Notice how, even while watching the public corkboard, you cannot find out that your friend is getting *your* money rather than someone else's; the transaction, while being public, is anonymous in the sense that the sender and the recipients are not linked.

To achieve this, Zerocoins uses a *cryptographic accumulator*, a structure used to test membership of elements, without keeping the list of members, along with *zero-knowledge proofs*, which allows you to prove your knowledge on some information, without sharing this information[13]. Combined, along with

[9] [An analysis of anonymity in the Bitcoin System - F. Reid, M. Harrigan]
[10] [De-anonymizing Social Networks - A. Narayanan, V. Shmatikov]
[11] March 2014
[12] As said, Zerocoins *doesn't* work with a single, trusted entity, however the analogy is made simpler that way, and the difference isn't conceptually interesting here.
[13] Imagine a discussion between two people, the first convincing the second of his knowledge of a specific number, without telling which number it is.

some other improvements, these two structures forms a Non-Interactive Zero-Knowledge Dynamic Accumulator, a structure that secretly hold information (here, coins), where information can be added, and with which one can prove that he knows one of the coin stored inside the accumulator, without telling information about that coin. This is exactly the behavior described with the corkboard and the voucher.

Concretely, when *minting* a Zerocoin, Alice create a Zerocoin with serial number S along with some knowledge $\pi$, adds it to the shared accumulator, along with its value in Bitcoins. The Bitcoins become frozen, only a proof of knowledge on a coin inside the accumulator can unlock them. To *spend* the Zerocoin, Bob runs the zero-knowledge proof with $\pi$ on the accumulator, effectively proving that he, or someone else (here, Alice), did add a coin $S$ in the past, along with some Bitcoins. At that point, some coins frozen along the accumulator are unlocked and sent to Bob.

Anonymity is guaranteed by the strong properties of cryptographic accumulator; with well-chosen parameters, guessing the members of an accumulator is as hard as the *discrete logarithm problem*[14], believed to be hard enough for everyday cryptographic use, and widely spread in various security schemes.

Zerocoins isn't perfect though; on one hand, you are forced to work with a fixed denomination of Zerocoins when minting and spending: think again about the corkboard example, if Alice is pinning a very specific amount, and at some later time, Bob ask to unpin the same very specific amount, it is highly likely that Alice sent the money to Bob. Therefore, you are restricted to work with 1$ bill, or 1 Zerocoins coins. This is cumbersome if you want to trade a lot of money. On the other hand, being forced to work with a fixed small amount allows another kind of attack: Suppose the Zerocoin network is idle (i.e. no one is minting). When trying to convert a certain (high) amount of Bitcoins, you will probably mint several Zerocoins successively, until you converted all your Bitcoins; then, the same problem arise: if someone, later on, ask to spend that exact number of Zerocoins together, it is likely that this person is linked to the one minting that specific amount of Zerocoins in a row.

## Final words

Where Bitcoins use a form of scrambling to improve the (bad) privacy protection offered to the users, Zerocoins uses cryptography and other proven scheme to achieve anonymity. Though not exempt of drawbacks, Zerocoin's functionalities is effectively a superset of Bitcoin's, and the part added, anonymity, is done in an elegant and simple manner, easy to review, as it is founded on a well-known, secure data structure.

[14] Computationally hard problem, based on the hardness of the discrete logarithm, i.e. given $a, b$, find $n : b^n = a$, which is equal to computing $\log_b(a)$. No polynomial-time algorithm exists for this problem.