WHITEPAPER

INDEX CHAIN

indexchain.org

Table of Contents

About Us

We are humbled by our exceptional community of supporters. We remain dedicated to innovating and driving blockchain adoption on a global scale. Our efforts significance and relevance are reflected in our ever-increasing community. The exponential inclusion of people from all walks of life has been an eye-opening experience. We welcome those seeking respite from the monopoly of banks wielding absolute power over their financial lives. Join us as we declare our independence and seek to restore our financial privacy.

Why Index?

INDEX CHAIN is designed for user privacy, shielding transactions with anonymous designations while deploying industry leading encryption methods. Index Chain is a complete solution, providing users with a fully private, secure, fast and decentralized solution. Protect your assets and remove banks from the equation. Avoid paying large sums with truly private transactions!

You do not have to fear about the blockage of your financial capabilities based on the whims of some power-hungry managers. Index Chain will allow you to become your own bank. You can spend your money safely and privately without leaving a trail of documents marking every step in your life.

Who are we?

We are a community dedicated to privacy. Comprised of consummate professionals with a passion for privacy, our goal is eschewing the age of bank control over personal financial situation and to establish the power of choice. INDEX is the ultimate solution, providing financial freedom coupled with opportunity.

Index Team

Our passion for privacy extends to the Index Chain Team. Comprised of the best of our exceptional community members, our team is given the choice to list their credentials. Intentionally optional, this approach reaffirms our dedication to maintain the choice of privacy in every facet of our operation. Those who establish themselves as trusted members of the community with regular contributions have the option to list their profile at the website - showcasing their skillset as well as the groups they are involved in.

Once again this is completely optional. For those interested we recommend reaching out to our team on discord. If you would like to have your profile added, contact us today to begin the process. Every member of the Index Chain Team exemplifies the following:

- **Professionalism** - Interactions are conducted in a professional manner with integrity
- **Communication** - clear and concise communication is evident, aiding others with helpful insight
- **Dedication** - remains steadfast in their convictions, displaying a tireless work ethic
- **Expertise** - exceeds the parameters of operational knowledge, is able to provide high level competency

MasterNode

The crypto community is a people-oriented estate where people come together to perform various functions in ensuring that the system keeps running at effectively as possible. There are lot of personality in a standard crypto network but we'll be expounding on what a crypto Node is and particularly what a Master Node is.

WHAT IS A NODE?

A crypto node is a personality on the cryptocurrency network that is charged with the responsibility of keeping all the statuses of the network in check. Each crypto network has its own set of modus Operandi and as such for the system to run effectively, the nodes are put in place to ensure that the model of operation agreed by the decentralized authority of the network are all kept by members and users of the network. One of the most common function of a node is to confirm transactions on the crypto network by solving an increasingly complex mathematical algorithm which grants them the right to create block and keep the chain rolling.

WHAT IS A MASTER NODE?

A master node is well different from an ordinary node on the network. A master node is a cryptocurrency full node that keeps a complete copy of blockchain happenings in real time. A master node also is responsible for other things such as

*Ensuring optimal security and privacy of the users

*Performing on-the-spot transactions

* More private transactions

*Taking part in setting the network standards

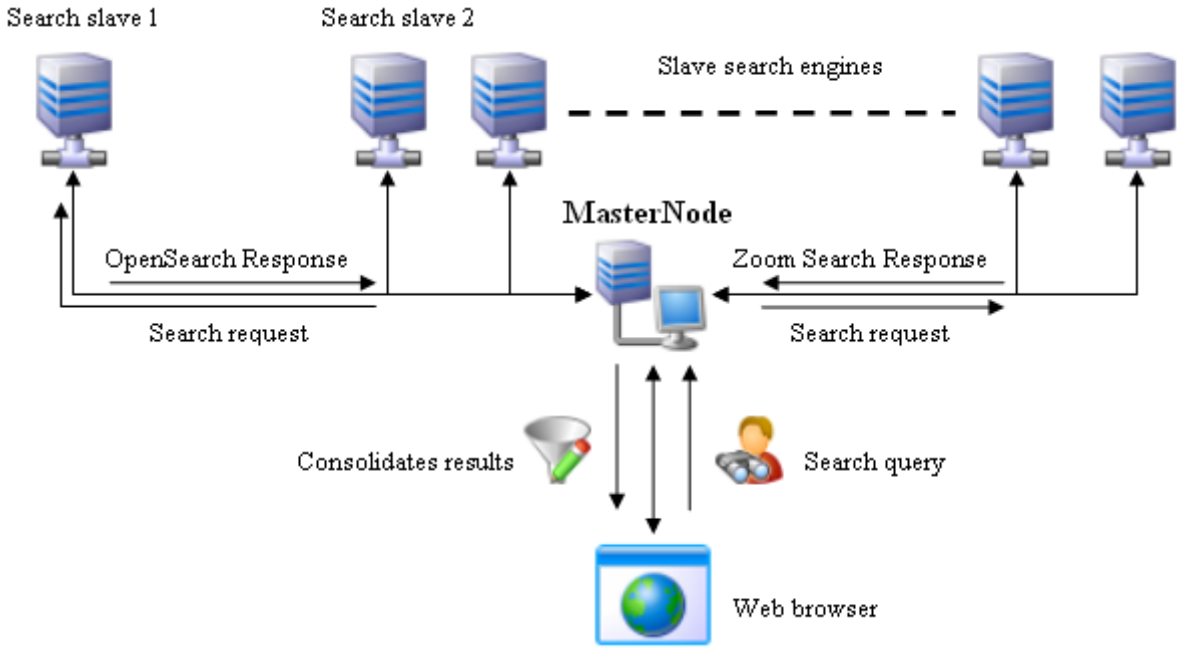
*Involved in decentralized voting systems, Masternodes stores all information about the network in wallets that are fully integrated and 24*7 connected with the blockchain network. Masternodes also verify or rejects new transactions that are added in the process of generating

a new block.
*Less energy consumptions, it is said that they are using 1500 times less energy

Master nodes are not singular in a crypto network but they also communicate with nodes like themselves on the decentralised framework. They are usually referred to as MN. It is however important to note that as much as the functions mentioned above represents the functions of a master node, there could be a little variance depending on the set up of each cryptocurrency.

RUNNING A MASTER NODE

Becoming a master node on any crypto network is not as easy as the entry requirement of being just an ordinary node. A master node is made to stake a huge percentage of their crypto on the network such that if there is any misuse of power, they can be punished by depreciating or depleting their stake.
Each masternode within the Index Chain blockchain needs a collateral of 5.000 IDX



Index Chain was developed as a hybrid currency with a higher reward system for Masternode aiming at the main growth point of the network; however, it is also optional to protect the network without the requirements of a Masternode by choosing other systems such as Proof of Stake 3.0 or Proof of Work -x16rv2. Because of the MN retention system, Masternode holders will receive a larger block reward due to the protection of the network through a lock guarantee

70% of the block's reward will go to Masternode holders. For example, if the block reward is 1 INDEX coins, 0.7 IDX will be rewarded to Masternode holders and 0.3 IDX will be distributed to the POW or POS 0.3 Coins. As more Masternodes are activated over time, more coins from the circulating supply will be blocked in the network. Enabling more Masternodes also helps to increase transaction speeds. With these variables, hyperinflation does not become a problem.

Reward:

To ensure the long-term sustainability of the IDX blockchain, its reward system was distributed to act in maintaining and expanding the network through a hybrid mining system divided between Proof of Work -x16rv2 and Proof of Stake 3.0 with 30 % in the rewards system and Masternode with 70%. This results in faster development and promotion, creating a virtuous cycle that benefits all stakeholders, including Masternode operators, Miners, Gamblers, investors and users.

A unique and sustainable system, with no unjust start, using a distributed hybrid system for block generation, eliminating high energy mining while being efficient and low maintenance costs, with a reward system controlled in a way to contain inflation and devaluation in the market in the long run.

As more miners, bettors and IDX Masternodes are activated, the network will become increasingly secure with faster transactions. Resistant to inflation, the hybrid reward system between PoW / PoS and Masternodes block currencies, divide the rewards into blocks and encourage the realization of 70/30 MN - PoW / PoS in divided rewards.

Proof of Work (Pow)

Proof-of-Work, or PoW, is the original consensus algorithm in a Blockchain network.

In Blockchain, this algorithm is used to confirm transactions and produce new blocks to the chain. Miners compete against each other to complete transactions on the network and get rewarded and create consistency in the blockchain data, prevents users from double-spending their funds, or attacking each other.

IndexChain uses the Proof-of-work (POW) mining method in its hybrid system. In crypto, the Proof-of-work Protocol or PoW is a protocol used for the prevention of cyberattacks such as DDOS and Spam. For this, a system was used where the user must prove that he has spent a certain time to find some answer that satisfies some requirement that the verifier asks for. Finding such an answer must be difficult and laborious for the protocol to work, but not impossible. The verification of this test, on the other hand, should be much faster and easier to be carried out, so as not to allow attacks.

This system emerged as an attempt to reduce the effects of these possible hash attacks used and its initial use was applied in the electronic correspondence system, announced in the summer of 1992, by Cynthia Dwork and Moni Naor at the 12th Annual Edition of the International Cryptology Conference , which took place from August 16 to 20, Santa Barbara, California, but was eventually applied to cryptocurrencies such as Bitcoin to verify transactions, secure consensus on the blockchain and mine coins.

Thanks to this implementation of the cryptocurrency system, Bitcoin could then be decentralized and have a secure algorithm to achieve Blockchain consensus and avoid situations like Double Spending or account manipulation, making it possible to confirm transactions and organize blocks in a way ordered. At Bitcoin, as a reward for PoW mining work, the system itself would reward miners with new Bitcoins until the 21 million bitcoins were generated for all those who

successfully solved the proof of work.

We can use a superficial view of PoW in the case of Bitcoin to exemplify the process: A new transaction is requested to be made. For this, it must pass the verification of function SHA-256. For a user to generate a proof to add to the order, he must ensure that the next block must have all N, first numbers of the hash result, the number zero. To generate this, it adds a cryptographic nonce to the end of the block, a nonce can be any numeric value that can only be used once. Nonce influences the result of hashing. The user tests nonce by nonce until he finds one that leads to the result with N, first numbers being zero. This can take thousands of attempts before finding a nonce that contains the answer. For all other users to check the validity of this next block, they only need to check it has the first twenty zero digits.

For a user to perform some action, he must be able to prove that he has performed a task, this proof system is the guarantee that the user has spent time to generate a response that satisfies some requirement of the evaluator. For this system to work, such proof must be laborious to create, but easily verified by the evaluator. The process involves taking the Encrypted Hash Function from the last block of the Blockchain (as in the case of Bitcoin where the SHA-256 algorithm is used), adding new transactions and solving a new encrypted function.

For regular users, the effort used to generate a proof is minimal, since he does not want to carry out a large volume of orders, but malicious users who intend to overload the system will be restricted by the time necessary to generate these proofs of work, causing no damage. In addition, if necessary, a new function can be adjusted to be faster or slower to generate a valid proof of work, according to what the recipient wishes.

There are two ways to apply this protocol. In a simple verification implementation, the user already has information about the task to be carried out and they only need to compute the result to send to the evaluator. The appraiser only needs to verify that the result fits the parameters before placing the order.This model is proposed for emails.

Another method is based on the container sending the tasks to the user. The user makes a request to the recipient and he, already knowing the possible answers, returns some task that the user must perform. Solved to the task, the result is sent by the user to the recipient, and if the result is any of the answers that the recipient knew, he accepts. The greatest interest of using this mode is the adaptability of the system, if necessary it can increase or decrease the difficulty of creating the proof, increasing or decreasing the time needed for the user to generate the proof, which is useful for mining in cryptocurrencies, as the case applied in the Index Chain's PoW mining system.

PoW is the original consensus algorithm on a blockchain network where this algorithm is used to confirm transactions and produce new blocks for the chain. With PoW, miners compete between to complete transactions and be rewarded. In a network, users negotiate other digital tokens with each other. A decentralized ledger gathers all the transactions of the blocks. However, care must be taken to confirm the transactions and organize the blocks. This responsibility is produced by special computers and by people called miners. The process is called mining.

The answer to the POW problem or the mathematical equation is called a hash. As the network grows, the difficulties of these equations increase more and more. Algorithms need more and more hashing power to solve the problem. Therefore, the complexity of the task is a delicate issue.

The precise work and speed of the blockchain system depends on it. But the problem shouldn't be too complicated. If so, block generation will take a long time. Transactions are stuck without execution and the workflow hangs for some time. If the problem cannot be solved within a defined period, the generation of blocks will be a kind of miracle. But if the problem is too easy, it is prone to vulnerabilities and there may be DoS attacks and spam. The solution needs to be easily verified. Otherwise, not all nodes will be able to analyze whether the calculations are correct. Then you will have to rely on other nodes or

violate one of the most important features of the blockchain - “transparency”.

This algorithm is implemented to the blockchain so miners solve the puzzle, form the new block and confirm the transactions. How complex a puzzle will be depending on the number of users, the current power and the network load. The hash of each block contains the hash of the previous block, which increases security and prevents any block violation. If a miner can solve the puzzle, a new block is formed. Transactions are placed in this block and are considered confirmed.

Proof of work is used in several cryptocurrencies. The most famous application of PoW is in Bitcoin. It was Bitcoin that laid the foundation for this kind of consensus. The puzzle is "Hashcash". This algorithm allows to change the complexity of a puzzle based on the total power of the network. The average block formation time is 10 minutes. Bitcoin-based cryptocurrencies, like Litecoin, have a very similar system. Another great project with PoW is Ethereum. Given that almost three of the four projects are implemented on the Ethereum platform, it is safe to say that most blockchain applications use the PoW consensus model.

The main benefits are the defense against DoS attacks and the low impact of participation in mining possibilities. Defense against DoS attacks - PoW imposes some limits on actions on the network. They need enormous efforts to be executed. The efficient attack requires a lot of computational energy and often to do the calculations. Therefore, the attack is possible, but it is useless because the costs are very high.

Mining requires highly specialized computer hardware to run the complicated algorithms. The costs are uncontrollable. Mining is becoming possible only for special mining plants. These specialized machines consume large amounts of electricity to run this cost increase. Large costs threaten the centralization of the system, since they benefit the government. This is quite easy to see with Bitcoin.

Miners must work hard to generate blocks and this consumes a lot of

energy. However, your calculations are not applicable anywhere else. They guarantee the security of the network, but they cannot be applied to business, science or any other field.

The main disadvantages are the huge expenses, "uselessness" of calculations and attack of 51%. A 51% attack, or majority attack, is a case when a user or a group of users controls most of the mining power. Hackers are given enough power to control most events on the network. They can monopolize the generation of new blocks and receive rewards, as they can prevent other miners from completing the blocks.

The 51% attack is not a profitable option. It requires an enormous amount of mining power. And, since all the activity is exposed publicly, everyone can become aware of everything, the network is considered compromised, which leads to the departure of users. This will inevitably bring down the price of the cryptocurrency. As a consequence, the funds lose their value.

Proof-Of-Work HASH X16R-V2

X16R algorithm was changed into X16RV2 by Ravencoin on the 1st of October 2019 to make it increasingly ASIC resistant while increasing the hash rate. Index also employs that strategy. To introduce X16RV2, the algorithm Tiger was introduced into three parts of the X16R algorithm. The Tiger hash is designed to perform before the three algorithms Luffa512, Kecca512, and SHA512. The previously used X16R used 16 different algorithms operating in chain fashion, and the ordering was dependent on the last 8 bytes of the hash of the previous block. The reason why the move from X16R to X16RV2 was made was that an ASIC mining machine was about to go online, an ASIC 4x16 – there are three different ASICs right now that are 4x16.

Another reason was the dark mining pools – the ones that are hidden from the eyes of the community or private and are giving the 36.7% unknown network hash rate – there are 36 blocks that are unknown to

the network right now. There are also secret FPGA crypto mining farms on X16R. Numerous issues are solved by the introduction of X16RV2.

-Support all video cards, while the latest version also hashrate for this algorithm is improved by 8-10% compared to previous version.

Proof-of-Stake v3.0

Proof of Stake was invented to solve many of these problems created by Proof of Work.

Proof of Stake's security has proven itself reliable & effective over

Coin	Worker	x16R 24 Hour	x16Rv2 (~5min)
Ravencoin	EVGA_gtx1070ti_8GB	20.53 Mh/s	17.56 Mh/s
Ravencoin	EVGA_gtx1060_SSC_6GB	12.19 Mh/s	10.48 Mh/s
Ravencoin	MSI_gtx1060_6GB	11.86 Mh/s	10.31 Mh/s
Ravencoin	EVGA_gtx1060_3GB	10.6 Mh/s	9.05 Mh/s
Ravencoin	HP_909616	10.33 Mh/s	8.47 Mh/s
Ravencoin	EVGA_gtx750ti	4.18 Mh/s	3.74 Mh/s

X16RV2 HASHRATE
COMPARED TO X16R

years of testing while at the same time solving Bitcoin’s issues caused by the Proof of Work (PoW) protocol. The latest Proof of Stake (PoS 3.0) have solved the issues faced with Coin-Age, Block Reward, Blockchain Precomputation. The PoS 3.0 protocol is now robust and keeps nodes connected to the network while disincentivizing inactive nodes

Proof of Stake

IndexChain use Proof-of-Stake (POS) as part of its mining method, but in particular, the most updated and improved version, PoS 3.0. Proof-of-Stake (POS) has proven to be reliable and effective over years of testing and, solving the problems of Bitcoin-derived systems caused by the Proof-of-Work (PoW) protocol. The latest advances in Proof-of-Stake are given the new version of PoS 3.0.
The Proof of stake cryptocurrency consensus mechanism explains that for any node to mine or create a block on the network, they must have a certain percentage digital asset.

For the cryptocurrency network to work effectively as a trustless decentralised that it is, then there must be a consensus mechanism that will be in place to guide the operations of every member in the network. On the cryptocurrency network, the personality that ensure that all model of operation are followed to the letter are called nodes. While there are various model of consensus mechanism that can be instilled in the crypto community in this article, we'll be expounding on the Proof Of Stake (POS).

Nodes or miners who are looking to make a fortune on the crypto network must provide services such as confirming transactions in real time that enables the network to work effectively. In the proof of stake mechanism, miners must themselves hold a significant amount of the coin on the network before that are awarded a block. This means that the higher the amount of digital asset a node has, the higher the percentage of possible blocks that can awarded to him. PoS was invented to curb the attack of miners on the crypto network with the belief that if the nodes have a stake in the network, there's a lower possibility of them attaching the network. It would simply be like stealing from themselves.

A huge amount of energy is needed to mine coin on the crypto network and instead of awarding the block based on the past glory of a miner, POS limits miners to mining only a certain percentage that is in proportion to their ownership stake.

To protect the blockchain network, there are two methods: the first is "Proof-of-Work (POW)" as we talked about previously and the second method is "Proof-of-Stake (POS)". The theory behind PoW is to maintain mathematical competition. The first computer to solve the puzzle confirms the transaction block, wins the currency reward. This is called mining. However, this creates problems of high wasted cost, high energy cost, high fees, slowness (slow transaction processing, slow tx / s) and centralizes the network on some sets of computers belonging to some rich people who could afford all hardware accounts, all because of the very nature of mining.

To compensate for this side created by PoW, we implemented PoS 3.0 in the Index Chain, which is an improved version of the original PoS and generates competition between coin holders, where, based on network connectivity and random chances, you can confirm a block transaction and receive currency rewards. This is called betting. It requires no specific hardware, except a normal computer with an internet connection, and you will be rewarded in proportion to the coins you have, which makes it fair and decentralized. Currency rewards are determined by annual supply inflation and awarded proportionally to the addresses of that share (equivalent to mining in PoS 3.0).

PoS 3.0 solves Bitcoin's PoW problems, as it is fast and low cost, while remaining decentralized. Below we will see the great security of PoS 3.0 and how it solves related security problems.

Security, coinage and attacks - The whole purpose of competing for coins is to avoid attacks. Confirming transactions is an honor given to a block winner. Although if this system can be used, it will be defective.

In PoS, you first prove that you have access to coins and, from that point on, you can compete to win blocks at random. The more people competing, the safer the block. The age of the coin is that the longer you hold coins, the more likely you are to win a block. Its original

intention was to encourage inactive coin holders. However, this does not encourage a node to remain connected to the network in practice, as you can expect the reward to increase. In addition, coin holders can disconnect from the network for long periods of time, reconnect and earn enough blocks to risk a 50% attack on the network. Calculating time will affect payments, discouraging connectivity. In addition, the fewer nodes are connected, the easier it will be to obtain most of the blocks that forge consensus. In addition, bets can be calculated in advance to make the attack more effective. Timestamps are used in PoS to get a general idea of the time. Deviation calculations are used to prevent falsifying incorrect time stamps.

In PoW, an increase or decrease in difficulty is made, depending on how quickly a block was produced. However, as a precautionary method to prevent any type of "Timing Attacks", the PoS development system uses centralized checkpoints.

All problems have a solution:

Age of the coin - The age of the coin is calculated by the weight of the unused coins and the time they have been inactive. The calculation is simply "proof of <currency · age · target". The proof hash is the hash of an obfuscation sum that depends on a stake modifier, the unspent output, and the current time. The attack to save Coin Age was previously described as unlikely. The reasoning behind this is because it is very difficult to make consecutive double expenses, as Coin-Age would be reset after the first expense. Although this is not entirely clear why an input can be divided into thousands of outputs. This can give the possibility of consecutive double-spending attacks. However, this is still a difficult problem because the attacker would need significant funds to maintain the weight greater than the network. In theory, this makes sense. Although if we look at the number of forks using PoS, we can see that the number of knots is quite low and this gives a much larger weight to a smaller handful of us. A holder of many coins may not want to carry out this attack, as they have the potential to lose the value of their coins if detected. As rational as this may seem, it is probably a fallacy, because it is still an attack vector

and, in fact, very vector. Most importantly, with so many currencies being published daily, keeping as many nodes connected as possible is essential for security.

Pre-computing in Blockchain - Block timestamp is essential for the PoS system. In theory, it is possible to fork a currency by changing the previous timestamps. The stake modifier does not obscure the hash enough to prevent knowledge of future evidence. Therefore, an attacker could try to calculate all the blocks in advance and be more likely to create several consecutive blocks.

PoS 2.0 solution: The bet modifier is changed at each modifier interval to better overshadow the calculations that would be made to identify the time for the next bet proof. The expected blocking time was increased from the original by 60 seconds to match the granularity.

C. Bulk reward - Unfortunately, the bulk reward on most PoS systems is based on the age of the coin. In theory, this is to distribute interest fairly, allowing nodes to receive latent payments due. It is an attempt to maintain a common APR. However, this system does not work because the nodes can remain disconnected and, with many split entries, reconnect to the network and play the reward system. In addition, it offers us no incentive to stay connected. In a decentralized system, the more nodes connected, the better the security, as it transfers the trust of a single entity to the network itself. PoS 3.0 solution: The block reward was made in 20 constant coins per block. This was proportional to the coin supply, maintaining interest at% 1.

Multi-signature and Stake Frio - The final noteworthy addition to the protocol was the implementation of "Multisignature Staking". A disadvantage of many stakeout algorithms is that they only support stakeout with a single key. Since the popularity and use of Bitbay, which uses a two-part guarantee system, also known as "Double Deposit Deposit" and extremely secure double key accounts, it has become important to allow these accounts to participate in protecting the network. Besides dual-key accounts, many other types of entries make use of lock and p2sh times, and these must also be allowed to

protect the network. The other problem is that, in a single key account, a hacker can use keyloggers to obtain your password and compromise your wallet while it is unlocked for stake application.

PoS 3.0 solution: Users could place the block signature key on output "6a", known as the recording address, so they can invest by sending a standard transaction.

This allows any entry to be eligible for submission. The "Cold Stake" technique involves several computers. Basically, when an entry with multiple signatures is eligible for stakeout, the signatures are split between many computers. This makes an account virtually impossible to hack because, even if a single key has been compromised, the other keys are in a completely different location, on the local network or on multiple servers. This technology is also already implemented in the Index Chain.

What is the difference PoS 3.0 to previous versions?

PoSv3 is really an incremental improvement over PoSv2. In PoSv2, the stake modifier also included the time of the previous block. This was removed to prevent a "short-range" attack, where it was possible to iteratively mine an alternative blockchain, repeating the previous blocking times. PoSv2 used block and transaction times to determine the age of a UTXO; this is not the same as the age of the coin, but the "minimum necessary confirmations" before a UTXO can be used to bet. This has been changed to a much simpler mechanism, where the age of a UTXO is determined by its depth in the blockchain. Therefore, this does not encourage inaccurate timestamps to be used on the blockchain and is also more immune to "timewarp" attacks. PoSv3 also added support for OP_RETURN cointake transactions, which allow a vout to contain the public key to sign the block without requiring a full payment script for pubkey.

- PoS 2.0 solution: Removing the minting from the equation -

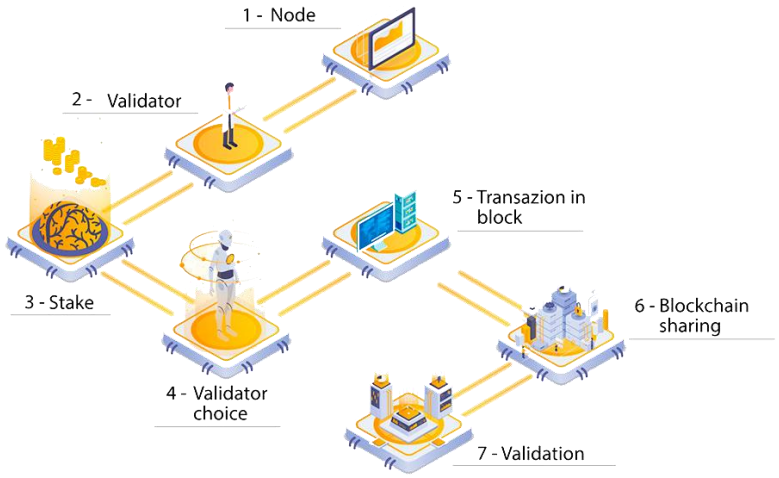
"proofhash <coins • target"

-The final noteworthy addition to the protocol was the implementation of the "Multisignature Stak

-We allow users to place the block signature key in output "6a", known as the recording address so they can bet by sending a standard transaction.

- Massive mining sets and centralizers are not required for PoS; anyone with a computer or cell phone can do it. Therefore, it would be even more decentralized, tending to randomization. Security would benefit and make access and energy efficient easier.

- Big players don't get disproportionately bigger rewards
-More computing power is not useful for creating blocks
-No network member can control the entire blockchain
pseudo-code:
while (blockhash> difficulty) {block.nonce = block.nonce + 1 blockhash = rx16v2 (rx16v2 (block)



This image shows validation process

Eliminating Block Reward based on time was an obvious improvement. Therefore, if the amount of stakeout of nodes falls, the annual interest for active nodes would increase proportionately. For example, if only 1/5 of the network was betting, you can expect up to 5 times the payoff! Unlike many PoS coins that do not have enough knots, this PoS 3.0 feature is a major advantage for small purses. Despite the lack of

statistical data on PoS currencies, there are generally less than 20% of participants who bet. In PoS 3.0, the aforementioned increase in incentive will keep the nodes more numerous, more competitive and, therefore, more decentralized. The change in granularity was useful to avoid "stake". Even with all the hashing power of the Bitcoin network, using PoS 3.0, a practice of attacking the network would be extremely unlikely to the point of being realistically impossible.

PoS 3.0 is one of the most secure and reliable systems ever created and Index Chain benefits greatly from this new system. Everything is done to guarantee anonymity, keep as many nodes connected as possible, guarantee decentralization and mitigate all attacks. Decentralization was Bitcoin's original core ideology, but unfortunately, Bitcoin's failures prevented it from prevailing eventually. The whole objective of a fair and secure financial system is to put control in the hands of people, so it is for people and for people. Fortunately, PoS 3.0 solved the main problems of Bitcoin's PoW and, at the same time, guarantees its own future by providing an ncentive to stay connected to the network to keep it safe and decentralized

Hybrid Pow + Pos Consensus Mechanisms

Consensus mechanisms are vital components that ensure the success of cryptocurrency’s. They meticulously regulate the transactions conducted within the systems of these currencies. This is needed because large volumes of transactions are recorded each day, and they require proper verification. These transactions are recorded in a ledger known as blockchains. Blockchains are not officially regulated, so other methods are needed to confirm the accuracy of the blockchains. If the blockchains contain the wrong data, the cryptocurrency has no validity or trust.

The blockchains need to be verified through the consensus of other systems. This is where PoW and PoS come in. Both of these mechanisms help protect the network they are assigned to and

prevent fraudulent activity. The data in the blocks needs to be correct for any cryptocurrency to have trust and legitimacy. Hybrid PoW + PoS ensure this and stop double-spending and other issues.

PoW data mining utilizes high-powered hardware to work through a mathematical problem known as a hash. Numbers are calculated from this solution and create a consensus. Good computer hardware can make this process more efficient. The better the hardware the better the calculations will be. Cryptocurrencies like bitcoin implement this method of consensus to ensure the value of their currency. PoW mining has its advantages, but it doesn't allow users to participate in the process of verification. This is why some networks turn to PoS mining.

PoS mining allows users to get involved with data mining. It does not have the same hardware requirements as PoW. Instead, blocks are created by individuals who have the largest stakes in the currency. This can be good in some ways. More people mine the currency, but this comes at the cost of equality. Those with more shares have an advantage over other users and will get more rewards.

Considering this, it is best if currencies implement a hybrid PoW+PoS. This gives users a fair chance to mine for data and currency. At the same time system improves on the security of these systems. No system is perfect alone but combined they have a better chance at maximizing the rewards for all users.

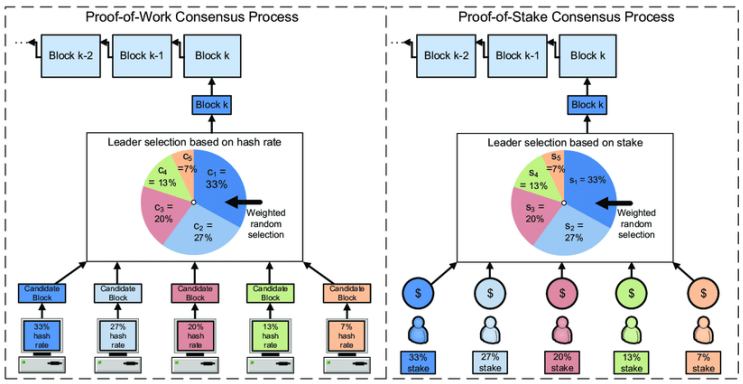
Proof of Work (PoW) x Proof of Stake (PoS)

The Proof of stake cryptocurrency consensus mechanism explains that for any node to mine or create a block on the network, they must have a certain percentage digital asset.

For the cryptocurrency network to work effectively as a trustless decentralised that it is, then there must be a consensus mechanism that will be in place to guide the operations of every member in the network. On the cryptocurrency network, the personality that ensure that all model of operation are followed to the letter are called nodes. While there are various model of consensus mechanism that can be instilled in the crypto community in this article, we'll be expounding on the Proof Of Stake (POS).

Nodes or miners who are looking to make a fortune on the crypto network must provide services such as confirming transactions in real time that enables the network to work effectively. In the proof of stake mechanism, miners must themselves hold a significant amount of the coin on the network before that are awarded a block. This means that the higher the amount of digital asset a node has, the higher the percentage of possible blocks that can awarded to him. PoS was invented to curb the attack of miners on the crypto network with the belief that if the nodes have a stake in the network, there's a lower possibility of them attaching the network. It would simply be like stealing from themselves.

A huge amount of energy is needed to mine coin on the crypto network and instead of awarding the block based on the past glory of a miner, POS limits miners to mining only a certain percentage that is in proportion to their ownership stake.



Proof of Work

Consider Bitcoin as an example of a cryptocurrency system secured with a proof of work algorithm. Each block in Bitcoin consists of two parts:

- block header of key parameters, including block creation time, reference to the previous block and the Merkle tree root of the block of transactions
- block list of transactions.

To reference a specific block, its header is hashed twice with the SHA-256 function; the resulting integer value belongs to the interval $[0,2^{256}-1]$. To account for different possible implementations, use a generic hashing function hash(·) with a variable number of arguments and range $[0,M]$. For example, arguments of the function can be treated as binary strings and merged together to form a single argument that can be passed to the SHA-256 hashing function.

The block reference is used in the proof of work protocol; in order for a block to be considered valid, its reference must not exceed a certain threshold:

$$\text{hash}(B) \leqslant M/D,$$

where $D \in [1,M]$ is the target difficulty. There is no known way to find B satisfying other than iterating through all possible variables in the block header repeatedly. The higher the value of D, the more

iterations are needed to find a valid block; the expected number of operations is exactly D.

The time period $T(r)$ for a miner with hardware capable of performing r operations per second to find a valid block is distributed exponentially with the rate r/D (see Appendix A):

$$P\{T(r) \leqslant t\} = 1 - \exp(-rt/D).$$

Consider n Bitcoin miners with hash rates r_1, r_2, …, r_n. The period of time to find a block T is equal to the minimum value of random variables $T(r_i)$ assuming that the miner publishes a found block and it reaches other miners immediately[1]. According to the properties of the exponential distribution, T is also distributed exponentially:

$$P\{T \stackrel{\text{def}}{=} \min(T_1, \ldots, T_n) \leqslant t\} = 1 - \exp\left(-\frac{t}{D}\sum_{i=1}^{n} r_i\right)$$

$$P\{T = T_i\} = \frac{r_i}{\sum_{j=1}^{n} r_j}.$$

;

The last equation shows that the mining is fair: a miner with a share of mining power p has the same probability p to solve a block before other miners. It can be shown that proof of work as used in Bitcoin satisfies Conditions 1–3.

Proof of Stake

In proof of stake algorithms, inequality is modified to depend on the user's ownership of the particular PoS protocol cryptocurrency and not on block properties. Consider a user with address A and

balance bal. A commonly used proof of stake algorithm uses a condition as

$$\text{hash}(\text{hash}(B_{prev}),A,t) \leqslant \text{bal}(A)M/D,$$

- B_{prev} denotes the block the user is building on,
- t is the current UTC timestamp.
-

For various reasons, some cryptocurrencies use modified versions of which we discuss in the corresponding sections.

Unlike , the only variable that the user can change is the timestamp t in the left part of equation. The address balance is locked by the protocol; e.g., the protocol may calculate the balance based on funds that did not move for a day. Alternatively, a PoS cryptocurrency may use unspent transaction outputs as Bitcoin does; in this case, the balance is naturally locked. A proof of stake protocol puts restrictions on possible values of t. For example, if t must not differ from the UTC time on network nodes by more than an hour, then a user can attempt no more than 7200 values of t. Thus, there are no expensive computations involved in proof of stake.

Together with an address A and a timestamp t satisfying (2), a user must provide a proof of ownership of the address. To achieve this, the user can sign the newly minted block with his signature; in order to produce a valid signature, one must have a private key corresponding to the address A.

The time to find a block for address A is exponentially distributed with rate bal(A)/D (see Appendix A). Consequently, the (2) implementation of proof of stake is fair: the probability to generate a valid block is equal to the ratio of user's balance of funds to the total amount of currency in circulation. The time to find a block for the entire network is distributed exponentially with rate $\sum a$ bal(a)/D.Thus, if the monetary supply of the currency $\sum a$ bal(a) is fixed or grows at a predictable rate, the dfficulty D should be known in advance:

$$D = \frac{1}{T_{ex}} \sum_a \mathrm{bal}(a)$$

,

with T_{ex} denoting the expected time between blocks. In practice, D needs to be adjusted based on recent blocks because not all currency owners participate in block minting.

Sigma Protocol

The Sigma protocol is a mechanism for proving that a statement or occurrence is true. It usually involves two participants; THE PROVER and THE VERIFIER. The prover's aim is to show that the statement or occurrence really is true without showing the verifier the key to understanding the statement or occurrence. The sigma protocol is close to the Zero-knowledge proofs. Although, the sigma protocol can stand alone, it is mostly used as a base for developing the zero knowledge proofs.

Sigma protocol involves a 3-round proof with the following:

1. Message from the prover to the verifier, expressing the fact that he has a truth and that he's willing to have it tested.

2. Challenge from the verifier with a random test to prove that the Prover actually can show the truth.

3. Proof provided by the prover to show that he actually knows and understand the truth without showing the Verifier how he did it.

In the event that the verifier is not satisfied with the fact that the Prover actually knows the truth - maybe the the Verifier thinks the prover just guessed right - then he can put another challenge to the Prover who has to answer by showing the truth and without showing the Verifier how it's done. This process can be carried out again and again with each challenge slimming the chances of the Prover to lie.

The Sigma privacy protocol represents a very important innovation in blockchain privacy, as it combines the high privacy of zero-knowledge proof schemes (ZKP), without many associated disadvantages, bringing great improvements to the IndexChain protocol. It provides an attractive alternative to zkSNARKs, with high anonymity and great performance, but it does so at the cost of reliable configuration, exotic encryption and complicated constructions. Sigma was originally introduced in the blockchain system as the next Zcoin replacement for Zerocoin. Sigma protocol has been introduced in the Index Chain structure to make significant improvements in relation to Zerocoin in

three areas:
- Reliable configuration removal
- Reduction of the test size from 25 kB to 1.5 kB
- Enhanced security

Sigma is based on the academic article One-Out-Of-Many-Proofs: Or how to leak a secret and spend a coin (Jens Groth and Markulf Kohlweiss) link: https://eprint.iacr.org/2014/764.pdf , which replaces RSA accumulators using Pedersen commitments and other techniques that cryptographic construction does not require reliable configuration. The only system parameters required in the Sigma configuration are the specifications of the ECC group and the generators in the group. This construction was further optimized in the Short Accountable Ring Signatures document, based on DDH (Jonathan Bootle, Andrew Cerulli, Pyrros Chaidos, Essam Ghadafi, Jens Groth and Christophe Petit) link: https://eprint.iacr.org/2015/643 .pdf that was used to further improve the construction.

Proof and safety sizes

Security through 256-bit ECC curves in Sigma is improved compared to the 2048-bit RSA used in Zerocoin and is estimated to equal the 3072-bit RSA. Our implementation of the Index Chain also uses the multi algorithms - Pippenger and Straus exponentiation for greater verification efficiency.

Trusted Configuration

Since the beginning of Zcoin, we have always seen the problem of “trusted configuration” as a major drawback. In a trusted configuration, some secret (public) parameters are generated based on a "primary private key". These network parameters are needed to create so-called "zero-knowledge proofs", which is the anonymity technology we use. The “primary private key”, sometimes called toxic waste, needs to be destroyed. If this data is not destroyed, someone with access to that key can generate an infinite amount of anonymous coins. One of the main criticisms of Zerocash and zkSNARKs (which

should not be confused with Zerocoin as used in Zcoin), as implemented in Zcash, is its requirement for having a reliable and controversial configuration.

An easy way to view a trusted configuration is to create a box with a lock on it and its corresponding key. Owning the key will allow you to create unlimited treasure from the box and therefore, the key must be destroyed. The trusted configuration effectively trusts that the key has been destroyed. But how do you know if it's destroyed? Unlike a physical object you can see, destroyed digital objects can always keep a copy or store it somewhere. Therefore, a basically reliable configuration means you need to trust someone or a group of people to destroy the key. If they didn't destroy it or if this ceremony was somehow hidden, someone has that key and can create money out of nothing. Sigma does not require this type of configuration because anyone who wants to help destroy part of the ring can participate.

Zerocoin, implemented by Zcoin, uses a reliable configuration performed by third parties in an academic challenge called RSA Factoring Challenge in 1991, where the incentive to insert a backdoor it is low and there was a considerable reward for breaking it. Although this is a decent implementation and with little chance of being compromised, we believe that the whole purpose of the blockchain is to build systems that do not require trust, and that same principle also applies to our privacy system. The initial launch of Zcoin in 2016 has been delayed, as our founder, Poramin Insom, spent many months trying to remove reliable configurations through the use of RSA UFOs, which proved to be impractical and had to settle for the parameters of the Factoring Challenge of RSA.

Enhanced security

Sigma's safety evidence is fully documented with much simpler construction, making it easier to audit. Sigma removes the reliable configuration and reduces the test sizes from 25 kB to 1.5 kB. The construction of Sigma does not suffer from the same flaw as the Zerocoin Protocol. The Sigma protocol allows users to prove that they

have complete privacy in transactions with no reliable configurations through zero-knowledge cryptocurrencies.

Zero-Knowledge Proof (ZKP)

The concept behind the zero-knowledge test is a unique method where a user can prove to another user he knows an absolute value, without transmitting additional information. Here, the tester can prove that he knows the X value for the verifier without giving him any information other than the fact that he knows the X value. The main essence behind this concept is to prove the possession of knowledge without revealing it. The main challenge here is to show you know a “y” value without saying what “y” is, or any other information.

If a user wants to prove a statement, he must know the secret information. In this way, the verifier could not transmit the information to others without actually knowing the secret information. Thus, the statement must always include that the taster knows the knowledge, but not the information itself. With that, you cannot say the value of "y", but you can say that you know "y". Here, "y" could mean anything.

This is the central strategy of applying the Zero-Knowledge Test. Otherwise, they will not be Zero-Knowledge Proof applications. That is why experts consider the applications of the Zero-Knowledge Test as a special case in which there is no chance to transmit any secret information.
The Zero-Knowledge test must have three different properties to be fully performed. They are:

Completeness - If the statement is really true and both users follow the rules correctly, the verifier will qualify the transaction with no outside help.
Solidity - If the statement is false, the verifier will not allow the transaction to take place in any scenario. (The method is checked to ensure that the probability of falsehood is equal to zero).
Zero Knowledge - The verifier does not store any information.

Exodus

We bring along the entire blockchain system of IndexChain, implementing the Exodus protocol, facilitating the use of smart contracts, personalized currencies/tokens and even decentralized exchange functions. This layer expands the utility and functionality of the IDX blockchain so as not to affect its core functions as a digital currency.

The Exodus protocol is a fork of the Omni protocol (Link: https://www.omnilayer.org/), best known for having Tether built into it. Exodus allows people to build our blockchain protected by an alternative PoW algorithm that, with the next MTP, will be resistant to ASIC.

Briefly, Exodus allows:
- People to create custom tokens on the Index Chain blockchain
- Blockchain-based crowdfunding
- Distributed exchange for decentralized trading of these tokens

Index Chain has implemented this layer of smart assets in its structure based on the structure implemented by Zcoin.

Dandelion ++

Dandelion ++ is a useful improvement over the original Dandelion protocol. Its integration for the launch of the Index Chain offers significant improvements in the privacy of the P2P network in the IDX network. Encryption attack vectors continue to evolve, as do solutions for them and Dandelion ++ represents another step forward in protecting user privacy applied to the large IDX system.

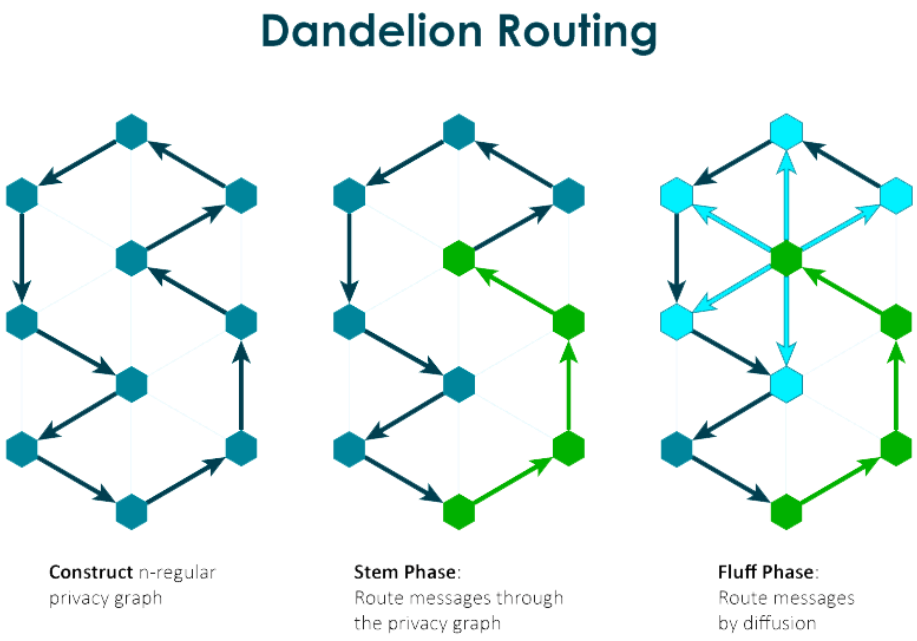
The Dandelion ++ protocol is an enhanced version of the Dandelion Protocol (which was originally proposed in 2017), to help improve the privacy of the Bitcoin P2P network. Dandelion ++ addresses concerns with the original protocol and has been implemented by the research team with a positive response from Bitcoin development teams.

Dandelion ++ is a direct network layer solution with anonymity being incorporated into the IDX network, explicitly enhancing the ideals of the original Dandelion proposal and differs from most broadcast communication anonymity protocols in addressing usage objectives and analysis metrics .

To understand how Dandelion ++ works, we must focus on how transactions are transmitted on the IDX network and how the original Dandelion protocol worked. In Bitcoin, when a user transmits a transaction from a node, it is propagated to the nodes connected to that specific node, known as its peers. The message of the transaction is then propagated in a chain reaction, in which each node spreads the message further to the nodes to which it is connected. This is known as the Bitcoin’s gossip protocol and is how transactions can reach most nodes on the network quickly.

Starting from the Dandelion ++ network, Index Chain implemented in its network a form of transmission known as diffusion, in which each node spreads transactions with exponential and independent delays to its peers, to mitigate the identification of the IP address of the network users.

Based on normal mapping, the origin of a transaction's sending and its IP address (which is not included in IDX transaction messages) can be mapped by observers if they control enough nodes or use a Supernode connected to many nodes on the network. They can actually map the source address by looking through which nodes the transactions comes from first. The role of Dandelion ++ is, precisely, to make a study of the network and identify it as a Supernode or a major node holder, recorded the traffic relayed from all P2P nodes and observe all patterns of transaction spreads overtime to, occasionally list the source IP address. By linking the IP address to the sender's alias, a third party can disarm users' names and link other transactions, even if a new public key is used for each transaction.
Dandelion was created to mitigate these vulnerabilities, but it had theoretical guarantees that did not apply effectively in practice. Dandelion's original proposal made three idealized assumptions on this basis:
- All nodes obey the protocol
- Each node generates precisely one transaction
- All Bitcoin nodes run Dandelion––



These assumptions did not work which is why Dandelion ++ tried to resolve them. The initial Dandelion protocol works in 2 phases:

1 - Stem Phase

2 - Fluff Phase

The Stem Phase is the anonymous system where the protocol reduces the possibility of mapping the network back to the IP address of the original node of the transaction message. In the Stem Phase, instead of a node sending a transaction to all connected pairs, it relays the transaction message through the privacy network to a single random pair based on an algorithm. Subsequent to the process, this node transmits the transaction message to another single point in the network, continuing the pattern until eventually (and randomly) a node transmits the message in a typical broadcast format to the rest of the network thus forming a web.

This is where the Fluff Phase begins to interact. After the last random node transmits the message using the network broadcast method, the IDX transaction message is then relayed to most nodes on the network quickly, thus making it much more difficult to trace the original node, as the transaction message it was passed on to many random nodes through a privacy graph before being propagated to allow the observer to map it to a single node on the network. Instead, an observer could map the propagation transactions back only to the nodes where the message was transferred in the Stem Phase, thus confusing the real identity of the sender of the transaction.

The Dandelion ++ protocol focuses especially on slightly changing the Dandelion implementation options, such as the graph topology and the message forwarding mechanisms within the network. Because of this operation, these small changes in the algorithm exponentially increase the space of the problem state for the analysis of anonymity. Dandelion ++ has increased the information that observers need to track to decode the users' names on the network.

Dandelion ++ differs notably from the original Dandelion in the Stem Phase, where it passes transactions through interlaced paths known as

cables before spreading the transaction message to the network. Interlaced paths can be fragmented, but their function when selecting a node to propagate the message is still confined to their local neighborhood. This is an important situation when comparing anonymity solutions at the network level, such as Tor, a routing protocol in which customers need current and global information from the network to determine the paths of transactions.

Dandelion and Dandelion ++ proceed in different cycles. Each node advances when its internal clock in the system reaches a certain limit. For each period, Dandelion ++ works on four main components, with small optimizations:

1 - Anonymity graph:

The anonymity graph uses the random 4-graph system (fig.2) instead of a linear graph system for the anonymity phase, with the choice of nodes whether or not their output neighbors support Dandelion ++ .

2 - Transaction routing (own):

Transaction routing (own) occurs whenever a node generates its own transaction, it forwards the transaction along the same output edge on the regular 4-graph. This differs from one of the problematic assumptions in Dandelion, in which the nodes are assumed only to generate a transaction.

3 - Transaction routing (elay):

Transaction forwarding (elay) is the probability moment in the stem phase when a node receives a stem transaction and retransmits or spreads the transaction over the network. The option to broadcast transactions to the network is pseudo-random. In addition, a node is a diffuser or a relay node for all retransmitted transactions.

4 - Fail-safe mechanism:
The fail-safe mechanism is the place where for each stem phase transaction, each node tracks whether it is seen again as a fluff phase transaction. Otherwise, the node broadcasts the transaction.

With these small adjustments in these stages of the algorithm, they make it more difficult to map IP addresses from the observation of propagation of transaction messages on the Index Chain network. The Dandelion ++ protocol continues to identify specific attack attempts that can be used against the original Dandelion protocol, including attempted graphics learning attacks, intersection attacks, graphics building attacks and black hole attacks. With each attack vector, they demonstrate how Dandelion ++ mitigates them with theoretical analysis and simulations.

Dandelion ++ does not increase the latency of the IDX network, and its practical feasibility has been demonstrated on the main Bitcoin network. It provides a lightweight and effective IDX network layer anonymity tool to reduce the possibility of mapping attacks to de-anonymize users. Despite its advantages, Dandelion ++ does not explicitly protect against opponents at the ISP or AS level, who can use routing attacks to discover a user's primary source on the network.

TOR Integration

TOR (initially The Onion Router) is an open-source software developed several years ago by the United States government, for the military, and later released for use by the population, TOR briefly creates encrypted “tunnels” of traffic overlying the internet, to provide privacy to the user.

The Tor community with the Crypto community, shared the ideal of privacy and decentralization. And in 2017, researchers from the University of Waterloo and the University of Concordia, both from Canada, introduced a system based on blockchain technology using onion routing techniques to facilitate anonymous deliveries.

Using the TOR network by the Index Chain protocol protects your IP address and the origin of the transaction with a deep level of anonymity, and as the number of IDX blocks increases, more nodes are added, which makes the network increasingly flexible and secure.

In a simplified way, the system works as follows:

Within the network, the TOR protocol finds an Entry Node in the network (or Entry Node) which is the initial connection node to the encryption protocol. The Entry Node is the place where IDX transaction data will enter the TOR network securely and anonymously. Between your computer and the Entry Node, a TLS (Transport Layer Security) tunnel is created. This tunnel is highly secure, no one can see what is going through it, all network traffic is encrypted from end to end. Will connect to another node within the network (Secondary Node) where a secure connection is established between two nodes creating a new cryptographic key (Key 2). There can be many Secondary Nodes in the network, the more flexible the network is. This Secondary Node connects to another node (Output Node) that will be where the data will leave the TOR network. The Secondary Node will then generate a new cryptographic key (Key 3) between it and the Outgoing Node, making sure that every transaction between them remains encrypted and secure.

Upon completion of all connections and data transaction between nodes, the Outgoing Node sends a request to its destination address, stating that all data has been individually encrypted by each Node. The server that received the request will know only that the request came from the Outgoing Node, but it will not be possible to track the route of connections and information exchanges traveled between other nodes in the network. Consequently you will not know where the initial transaction was sent from.

The final result obtained within the chain is that each node will know only the request sent through the node before its connection and the Login Node (which is the initial connection node) recognizes only your computer but does not know the destination of the data. This way, the network encodes its IP addresses between different connections, making tracing or identifying the principle of the transaction invisible.

Comparison Dandelion++ and TOR System

Tor's integration at the network layer level of cryptocurrency systems is extremely challenging. Monero is an excellent example of this, as it took four years to implement his Tor-like I2P Kovri project on his network and it is still a work in progress. Many cryptographic networks do not have the time or technical knowledge to integrate this functionality into their system.

Users who transmit their transactions via Tor, are not viable for ordinary network users like Bitcoin, unaware of the privacy deficiencies of the network or do not have the experience necessary to transmit transactions via Tor properly. In addition, the Tor system can be slow due to limited bandwidth compared to the Dandelion ++ protocol.

In addition, studies have identified concerns about bitcoin spreading animation. It also highlights attacks on us where they reject or blacklisting Tor network connections. This can lead to transaction cancellation and mapping of users' IP addresses making them vulnerable in the network.

Zero-Knowledge -Definition

Definition 1 (Zero Knowledge) *Let* (P,V) *be a interactive proof for* $L \in$ NP*, with witness relation* R_L*.* (P,V) *is* zero knowledge *if for all probabilistic polynomial time machines* V^* *there exists an expected PPT S such that for all nonuniform PPT D there exists a negligible function* ε *such that* $\forall x \in L, w \in R_L(x), z \in \{0,1\}^*$*, D distinguishes the following distributions with probability* $\varepsilon(|x|)$*:*

$$\{View_{V^*}[P(x,w) \Leftrightarrow V^*(x,z)]\},\{S(x,z)\}.$$

Perfect zero knowledge is exactly the same except that it requires the two distributions to be identical rather than simply indistinguishable.

1. An alternative definition is to replace $V\ IEW_{V^*}$ with $OUTPUT_{V^*}$. The two definitions are equivalent, since the output is included in the view and since V^* could simply output its view.

Definition 2 (Zero Knowledge) *Let* (P,V) *be a interactive proof for* $L \in$ NP*, with witness relation* R_L*.* (P,V) *is* zero knowledge *if there exists an expected PPT S such that for all probabilistic polynomial time machines* V^* *and for all nonuniform PPT D there exists a negligible function* ε *such that* $\forall x \in L, w \in R_L(x), z \in \{0,1\}^*, r \in \{0,1\}^*$*, D distinguishes the following distributions with probability* $\varepsilon(|x|)$*:*

$$\{V\ iew_{V^*}[P(x,w) \Leftrightarrow V^*(x,z)],r\},\{S^{Vr_*(x,z)}(x,z),r\}.$$

Definition 3 (Commitment Scheme) *Com is a* commitment scheme *if Com is polynomial time and there exists a polynomial* $`$ *such that the following two properties hold:*

Hiding: For every nonuniform PPT D there exists a negligible function ε *such that for all* $n \in$ N*,* $v_0, v_1 \in \{0,1\}^n$*, D distinguishes the following distributions with probability at most* $\varepsilon(n)$*:*

$$\{r \leftarrow \{0,1\}^{`(n)} : Com(v_0,r)\},\{r \leftarrow \{0,1\}^{`(n)} : Comm(v_1,r)\}.$$

Binding: For all $v_0, v_1 \in \{0,1\}^n$*,* $r_0, r_1 \in \{0,1\}^{`(n)}$*, if* $v_0 \neq v_1$ *then* $Com(v_0,r_0) \neq Com(v_1,r_1)$*.*

Commitment schemes can be constructed from OWPs (or OWFs):

Lemma 4 *If one-way permutation exist, then there exist (perfectly binding) commitment schemes.*

Proof. We begin by constructing a single-bit commitment scheme. Let f be the assumed one-way permutation, and let h be a hard-core predicate for f. We define a commitment scheme by:

$$Com(b;r) = (f(r), h(r) \oplus b).$$

Zero-Knowledge -Proof

Proposition 1 *If* (P,V) *is a ZK protocol, then* (P,V) *is witness indistinguishable.*

Proof. By definition of *ZK*, there exists a simulator *S*, such that:

$$\{P(x,w_1) \boxed{\leftrightarrow} V^*(x,z)\} \approx \{S(x,z)\} \approx \{P(x,w_2) \boxed{\leftrightarrow} V^*(x,z)\}$$

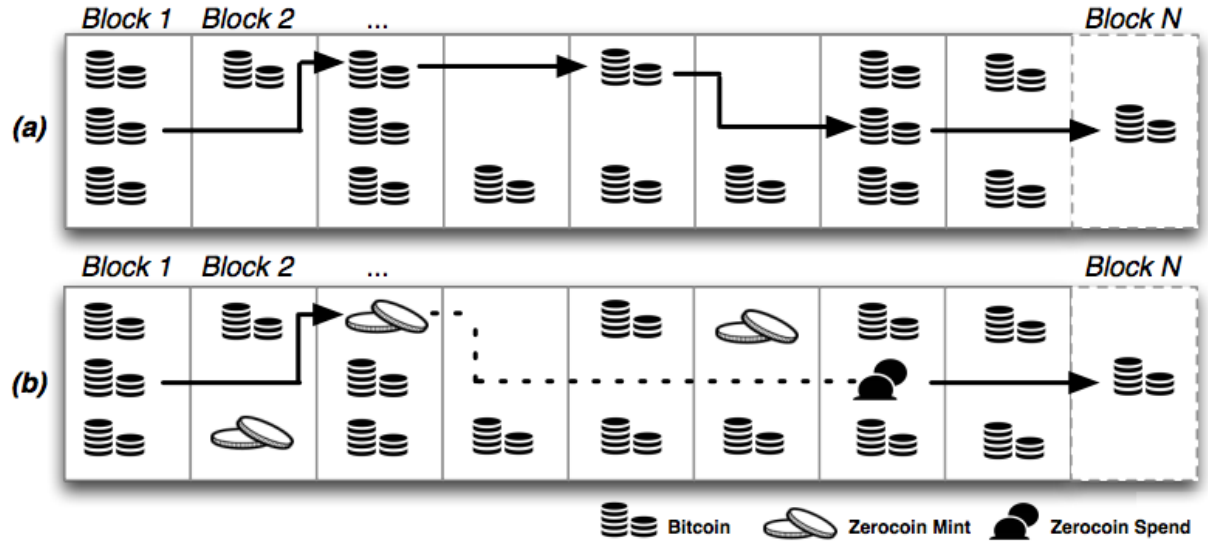
By the hybrid lemma, $\{P(x,w_1) \boxed{\leftrightarrow} V^*(x,z)\} \approx \{P(x,w_2) \boxed{\leftrightarrow} V^*(x,z)\}$. So (P,V) is witness indistinguishable.
Finally, here is the theorem that says why Witness Indistinguishability is nice to work with.

Theorem 6 *If* (P,V) *is WI, then* (P^n,V^n) *is WI. In other words, WI protocols can be repeated polynomially many times in parallel and still be WI.*

Proof. We want to show that $\{P^n(x,w_1) \boxed{\leftrightarrow} V^{*n}(x,z)\} \approx \{P^n(x,w_2) \boxed{\leftrightarrow} V^{*n}(x,z)\}$. To do this, define the following hybrids: Let H_i denote the view where the prover uses w_1 for the first i executions of the protocol, and w_2 for the rest. Then it is clear that our problem is exactly showing that $H_0 \approx H_n$. If they were distinguishable, then by the hybrid lemma, some $H_i \not\approx H_{i+1}$. However, I claim that this is not possible by efficient operations. Because the prover is efficient, we can efficiently simulate the entire protocol where the prover uses w_1 and output the view. We can do this i times. Likewise, we can efficiently simulate the entire protocol where the prover uses w_2 and output the view. We can do this $n - i - 1$ times. So concatenation by views of a *WI* protocol for a fixed witness is an efficient operation. Now observe that H_i is exactly $\{P(x,w_1) \boxed{\leftrightarrow} V^*(x,z)\}$ with i copies of the view of $\{P(x,w_1) \boxed{\leftrightarrow} V^*(x,z)\}$ pre-concatenated, and $n - i - 1$ copies of the view of $\{P(x,w_2) \boxed{\leftrightarrow} V^*(x,z)\}$ concatenated. Next, observe that H_{i+1} is exactly $\{P(x,w_2) \boxed{\leftrightarrow} V^*(x,z)\}$ with i copies of the view of $\{P(x,w_1) \boxed{\leftrightarrow} V^*(x,z)\}$ preconcatenated, and $n - i - 1$ copies of the view of $\{P(x,w_2) \boxed{\leftrightarrow} V^*(x,z)\}$ concatenated. So because (P,V) is a *WI* protocol, $\{P(x,w_1) \boxed{\leftrightarrow} V^*(x,z)\} \approx P(x,w_2) \boxed{\leftrightarrow} V^*(x,z)\}$, and by efficient operations, $H_i \approx H_{i+1}$. So we cannot have any $H_i \not\approx H_{i+1}$, so we have $H_0 \approx H_n$, and (P^n,V^n) is also a *WI* protocol.

How ZeroCoin Work

Zerocoin allows direct anonymous payments between parties. Zerocoin transactions exist alongside the (non-anonymous) Bitcoin currency. Each user can convert (non-anonymous) bitcoins into (anonymous) coins, which we call zerocoins. Users can then send zerocoins to other users, and split or merge zerocoins they own in any way that



preserves the total value. Users can also convert zerocoins back into bitcoins, though in principle this is not necessary: all transactions can be made in terms of zerocoins.

What makes Zerocoin and the new Zerocash protocol different from previous approaches:

Zerocoin and the Zerocash protocol operates in the Bitcoin network and is implemented as a series of extensions to the existing Bitcoin protocol. This approach means that Zerocoin can be deployed without relying on a central coin issuer or bank (as used in previous e-cash schemes). Moreover, since no single trusted party operates the Zerocoin system, attacks on Zerocoin must take on a substantial fraction of the Bitcoin network.

The Zerocash protocol uses provably secure cryptographic techniques to ensure that Bitcoins cannot be traced. These techniques allow users to conduct transactions on the Bitcoin network while receiving strong mathematical guarantees that the transactions cannot be traced. These guarantees remain in place even if a portion of the Bitcoin network is compromised by an attacker.

Other anonymous cash systems rely on distributing the work of anonymizing users amongst a set of parties. This approach works well if all parties are fully available but can be subject to "denial of service" attacks where a small number of nodes are taken offline. Because Zerocoin is built on top of Bitcoin, it is widely distributed among all the Bitcoin peers, ensuring that the system can remain available even when many nodes are compromised.

notation	example	meaning
`monospace`	`Encrypt`, `challenge`	algorithm/procedure/oracle names
$\leftarrow$	$y \leftarrow f(x)$	assignment
$\rightarrow$	$f : X \rightarrow Y$	function definition
$\twoheadleftarrow$	$b \twoheadleftarrow \{0,1\}$	uniform random sampling
$\twoheadrightarrow$	$\texttt{Encrypt} : PK \times M \twoheadrightarrow C$	randomised algorithm
$\bot$		a special symbol denoting "failure"
$\uplus$	$M \uplus \{\bot\}$	disjoint union (coproduct)
$\mathbb{N}$		natural numbers, including 0
$[\,]$		empty list
$::$	$L \leftarrow L :: l$	append to list

Dark Gravity Wave – What is DGW? | Difficulty Retarget Algorithm

In most of our mining guides we've shared coins that has difficulty retarget algorithm as Dark Gravity Wave. People often ask what is Dark Gravity wave and how this difficulty retarget algorithm works. If you are following Coin Guides then you'd know that when we give introduction to a coin we'll first share its technical specifications. Only then we'll get into wallet tutorial or mining guide. While we've explained everything in and out about a coin we never explained any of its technical features; what it is or how it works. For beginners to understand Bitcoin, Blockchain and Cryptocurrency terms we opened this new section called knowledge base where we are only going to write about technical terms. This is the first post in knowledge base and it's about Dark Gravity Wave.

What is Dark Gravity Wave (DGW)?

In Bitcoin and other Cryptocurrencies "Dark Gravity Wave" is an open source mining difficulty re-adjustment algorithm developed by Evan Duffield (creator of X11/Darkcoin/Dash). The first Crypto currency to implement this algorithm is Darkcoin DASH or Digital Cash. Later on; many Alt coins started to adopt this algorithm as it is known to adjust the difficulty faster and its non-linear.

Before DGW came KGW (Kimoto Gravity Wall), a most popular difficulty re-target algorithm that adjusts difficulty every block using information from the previous blocks. Dark Gravity Wave was inspired

and is based on Kimoto Gravity Well (KGW). Also DGW is proven to reduce some theoretical disadvantages of KGW such as time-warp exploit. Also there are 2 other difficulty adjustment algorithm namely Nite's Gravity Wave and Digishield. You'll come across all these algorithms only in Proof-of-Work mineable coins and not on Proof-of-Stake coins.

Why DGW and How it works?

In Bitcoin the standard block difficulty readjustment is set to adjust only every 2016 blocks. The problem with this scheme is that it gave rise to multipool mining. Multipool mining is a process of jumping from one crypto to another mining the most profitable one at that current moment. Then the miners dump the mined coins to buy back Bitcoins. True, this actually happened back then when the price of Bitcoin Cash (BCH) arose .

Miners will only focus on economic incentives; as BCH became more profitable miners almost abandoned Bitcoin network to mine BCH. Once BCH adjust its difficulty miners will then jump back to mine Bitcoin. People actually thought its 51% attack but it's actually nothing but a seesaw of hashing power being delivered between Bitcoin and Bitcoin Cash based on their profitability. This was a serious problem with Bitcoin and this is what gave birth to Dark Gravity Wave and other mining difficulty regulator

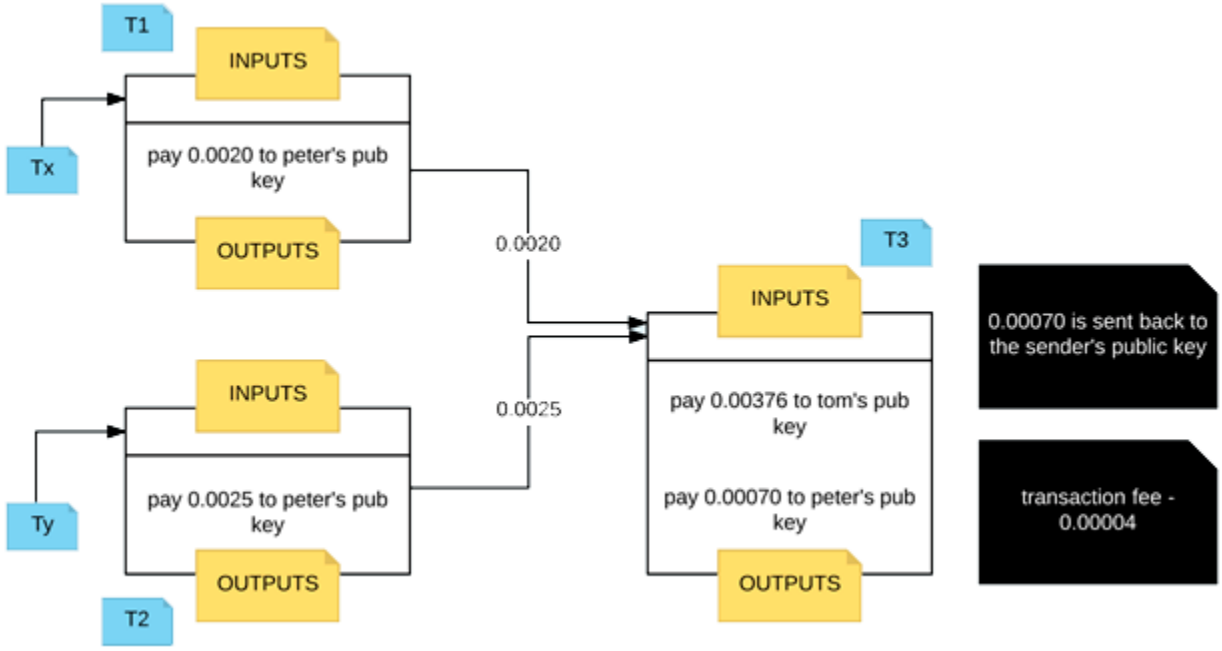
Benefits of Dark Gravity Wave

DGW uses multiple exponential moving averages and a simple moving average to achieve the smoother difficulty re-target mechanism. Coins that have Dark Gravity Wave as their difficult algorithm are immune to

issues like multipools as it retargets difficulty every single block. Not just that; with DGW the chain becomes more secure and block times are much more consistent; despite large fluctuations in mining power. Apart from controlled difficulty some other benefits of Dark Gravity Wave are security, faster transaction, more miners and reliable chain. Hope this information helps!

Bitcoin Transaction- technical explanation

The following images show Bitcoin transaction:



How does a bitcoin transaction work?

So if I'm Bob and I want to pay Alice, those inputs are my proof that I have been given a certain amount of money (although this might just be a portion of my total balance), and the outputs will correspond to Alice's account.

Main

hash: The hash over this entire transaction. Bitcoin generally uses hash values both a pointer and a means to check the integrity of a piece of data.

ver: The version number that should be used to verify this block. The latest version was introduced in a soft fork that became active in December 2015.

vin_sz: The number of inputs to this transaction. Similarly, vout_sz counts the number of outputs.

lock_time: Describes the earliest time at which a block can be added to the blockchain. It is either the block height or a unix timestamp.

Input
previous output hash: This is a hash pointer to a previously unspent transaction output (UTXO). Essentially, this is money that belongs to you that you are about to spend in this transaction.
n: An index into the list of outputs of the previous transaction. This is the actual output that you are spending.

scriptSig: This is a spending script that proves that the creator of this transaction has permission to spend the money

Output
value: The amount of Satoshi being spent (1 BTC = 100,000,000 Satoshi).

scriptPubKey: The second of two scripts provided in a bitcoin transaction, which points to a recipient's hashed public key.

Transaction verification

The function bitcoin node is the verify that incoming transactions are correct (data hasn't been tampered with, money isn't being created, only intended recipients spend UTXOs, etc).

All outputs claimed by inputs of this transaction are in the UTXO pool. Unspent outputs can only ever be claimed once.

The signatures on each input are valid. More precisely, we're saying that the combined scripts return true after executing them one after the other. More on this in the last section.

No UTXO is spent more than once by this transaction. Notice how this is different than the first item.

All of the transaction's output values are non-negative.

The sum of this transaction's input values is greater than the sum of its output values. Note that if the numbers are different, the difference is considered to be a transaction fee that can be claimed by the miner.

In the example, you can find a transaction ID highlighted in yellow. Meta-data along, with a description, is on the right. Inputs and outputs are highlighted in pink and green

Bitcoin Transaction- simple explanation

To understand how a bitcoin transaction works, it is important to understand what bitcoin itself is. Bitcoin ordinarily is an intangible digital asset that was created to carry out the functions of fiat currencies and more. For example, the 'exchange of value' function of fiat. Also, bitcoin is not a single unit coin but more like a file (which

represents money) that is moved when a payment or receipt transaction is initiated.

There are three major components of every bitcoin transaction and that includes: The Input, The Output and the registered amount.

The Input

The input transaction represents the address or source of the bitcoin. Such that for every collection of bitcoin unit that is transferred from one source to another, an address of where it originated from is stated. This ensures that every single movement of the littlest amount of bitcoin all goes into a proper immutable record including where they came from.

The Output

The output transaction is simply the other end of the input transaction. The output represents and codifies all necessary information about the receiver of the bitcoin. To receive any amount of bitcoin, you'll have to generate a receiving or output address from your end and send that address to the person who is going to initiate the sending or input transaction. The sender then copies your unique receiving address and initiates that the bitcoin be sent to that address. The output address is more like your bank account number for receiving funds. It is important to note that a single tweak in an output address makes it totally unrecognized in the bitcoin network.

The Amount

For every transaction of bitcoin made, there is a deliberate effort made to enter how many unit of bitcoin is sent or received. It is possible to receive a tiny fraction of bitcoin in a transaction while it is also possible to send huge numbers of bitcoin in a transaction. The system is built in such a way that the bitcoin can be broken down beyond the unit of one (1) meaning that you can send 0.5 bitcoin or even 0.005 bitcoin.

Bitcoin Transaction

When you send bitcoin to someone, your address is saved on the bitcoin network relating to that amount of bitcoin you sent. So also, when you receive bitcoin from another party, your address is stored on an inaccessible record. The record of transactions to and fro enables every unit of the bitcoin in circulation to be accounted for. In fact, it means that if transactions are to be traced, we can do a genealogy research of who made the very first bitcoin transaction and to whom it was sent.

HOW AND WHY BITCOIN ADDRESSES ARE CHANGED AFTER EVERY TRANSACTION.

The bitcoin network is a system and so is the entire unit of bitcoin in the network. The way the system was built makes us know that all units bitcoin available on the network is like a loaf of bread. When you share out of this loaf, you give it an entirely different identity when it reaches its destination. More importantly, a new address is generated to incorporate the bitcoin unit you sent and the units the receiver has in his wallet before yours joined. These are two loafs, from the same origin coming together, again. And they take another identity and become a whole.

Just like when you send $5000 dollars to someone who has $2000 IN their bank account before. When the transaction is completed, the beneficiary will have a while $7000 IN their account even though there is a record of how the $7000 came to being.

Also, after every transaction from the sender, the remaining bitcoin balance he has generates a new address. To ensure this, when sending a bitcoin unit, a sender is made to send the whole unit of bitcoin he has and then the bitcoin network then spilt it within the sender and the receiver. For instance, if I have 1 bitcoin and I want to send 0.5 of my bitcoin to someone, once I initiate a 0.5 transaction, my entire one bitcoin is lifted and divided into two and one part is sent to the other back to the senders wallet

Specification

NAME: INDEX CHAIN
TICKER: IDX
ADDRESS LETTER: i
ALGORITHM: x16rv2
MAX SUPPLY: 350.000.000
BLOCK TIME: 60 Seconds (1 minute)
BLOCK TYPE: Hybrid PoW/PoS/MN
MASTERNODE COLATERAL: 5000 IDX
BLOCK REWARD: 1 Coin Per BLOCK.
MASTERNODE - 70% Reward (0.7 coins per Block)
POS – 15% Reward (0.15 ~ 0.30 coins per Block)
POW- 15% Reward (0.15 ~ 0.30 Coins per Block)
COINBASE MATURITY: 100 Blocks
STAKE MATURITY: 100 MIN
MASTERNODE CONFIRMATIONS: 15
MASTERNODE MATURITY: 30 min
P2P PORT: 7082
RCP PORT: 8888

REWARD BLOCKS

--1° year Reward Block:	1.00 Coin	15% PoS	15% PoW	70% MN
--2° year Reward Block:	0.80 Coin	15% PoS	15% PoW	70% MN
--3° year Reward Block:	0.64 Coin	15% PoS	15% PoW	70% MN
--4° year Reward Block:	0.51 Coin	15% PoS	15% PoW	70% MN
--5° year Reward Block:	0.40 Coin	15% PoS	15% PoW	70% MN
--6° year Reward Block:	0.32 Coin	15% PoS	15% PoW	70% MN
--7° to 25° year Reward Block:	0.27 Coin	15% PoS	15% PoW	70% MN

Conclusion

In this whitepaper, we present a new hybrid and private cryptocurrency scheme that meets the requirements of a good privacy protocol, that is, a set of high anonymity, minimum confidence required, scalability, ease of use and implementation. Bringing formal proof of security and an entire structure showing its cryptographic construction base used in our Index Chain system.

In summary, the Index is a structure that is based on the basic Bitcoin protocol and incorporates several improvements and additional technologies in that protocol based on the assigned structures of Zcoin to make it perfectly anonymous, safe, economical and efficient for all users.

A wide variety of technologies have been used in their most recent and updated versions, and all flaws in Bitcoin's infrastructure in relation to security and anonymity have been properly addressed. A hybrid PoS and PoW (X16RV2), Sigma Protocol, Dandelion ++ technology, Masternodes, Lelantus and much more have been unified in an absolutely anonymous currency - Index (IDX).