# Darkcoin: Peer-to-Peer Crypto-Currency with Anonymous Blockchain Transactions and an Improved Proof-of-Work System

*Evan Duffield, Kyle Hagan*
*(evan@darkcoin.io, kyle@darkcoin.io)*

*18 March 2014*

## Abstract

Darkcoin is the first privacy centric cryptographic currency based on Satoshi Nakamoto's Bitcoin. DarkSend, a technology for sending anonymous block transactions is incorporated directly into the client using extensions to the core protocol. An improved proof-of-work using a chain of hashing algorithms replaces the SHA256 algorithm and will result in a slower encroachment of more advanced mining technologies (such as ASIC devices). DarkGravityWave is implemented to provide quick response to large mining power fluctuations.

## Introduction

Bitcoin was a remarkable invention. The concept of proof-of-work allowed, for the first time, decentralized consensus on a large scale network with no central authority. However, due to the very nature of decentralization, the blockchain is inherently not private. This has obvious implications for users' personal privacy, as all transactions are traceable in the block chain.

To solve this inherent problem of privacy, we created a new cryptocurrency: Darkcoin.

Darkcoin uses a decentralized implementation of CoinJoin in order to anonymize transactions. We named this implementation "DarkSend".

## DarkSend

DarkSend is a CoinJoin-based, decentralized peer-to-peer being implemented into Darkcoin. DarkSend provides protocol extensions to merge transactions together into larger anonymous transactions. This system uses regular nodes and elects a master node to create the transaction in a decentralized fashion.

DarkSend is a completely trustless solution, where users can achieve a high level of anonymity. With the exception of a collateral transaction (which will be explained in detail later), users run no risk of losing any money at any time.

This implementation of DarkSend is optionally available through the client and can be deactivated at any time if a user wishes.

DarkSend is available as an option through the client and can be deactivated by the user at any time. The DarkSend implementation gathers the required transaction information in multiple stages within each session:

- accept inputs and blinded outputs
- accept outputs
- elect a master node
- broadcast the finalized transaction
- sign
- collect or destroy collateral

A scheme using blind signatures is implemented to prove the provided outputs belong to one of the participants of the pool. Using this strategy neither the master nor other nodes know which outputs belong to which inputs.

**Defending Against Attack**

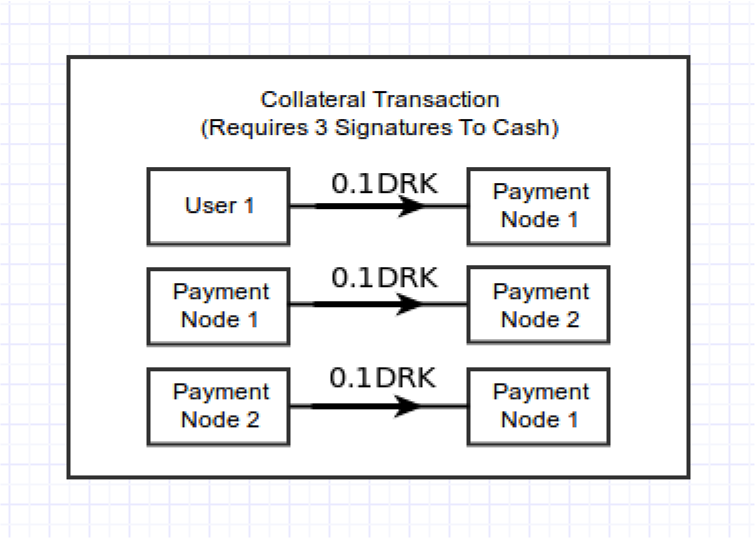
With the decentralized implementation of DarkSend, there are inherent challenges to dealing with rogue users who intend (or attempt) to attack the system. Such users could modify the software in a way that would cause it to refuse to sign, which would force the pool to reset every session.

To defend against various attacks, DarkSend implements a collateral system. A transaction for 0.1DRK is made out to the payment node to ensure proper usage of the system. This transaction is separate from the funds added to the DarkSend pool. If a user submits an input but refuses to sign or leaves at any stage, the payment node will “cash” the transaction by signing and broadcasting it. Collateral transactions require multiple signatures to complete from more than one payment node.

Payment nodes are simply the last node to create a block - specifically, the last block solver and the one before that. These nodes will monitor DarkSend for misbehavior. Should any be discovered, the payment nodes will “cash” the transaction by signing and broadcasting it. This has the added benefit of creating a sustainable income stream - in addition to mining - for miners, while simultaneously protecting the network from attackers.

The collateral transaction is made to multiple payment nodes. Cashing collateral transactions require multiple signatures from the user, payment node 1 and payment node 2.

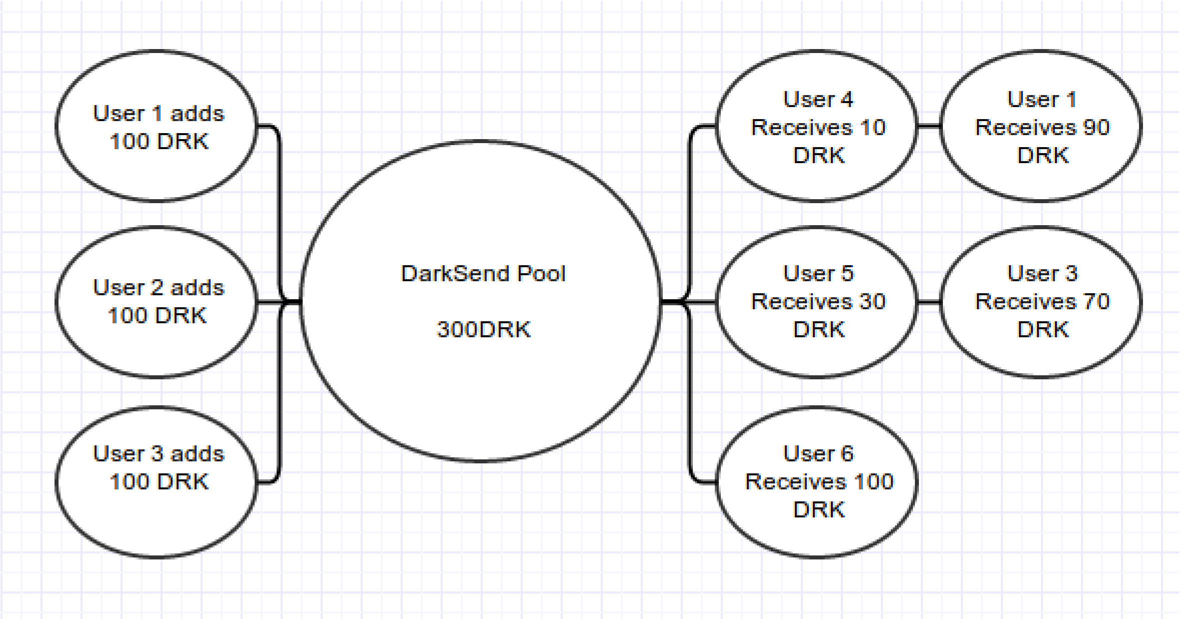
Collateral forfeited to the network will be paid to the payment nodes, which are the last two nodes to solve a block. These nodes will commonly be the pools. To cash the collateral transactions and take all of the money, multiple pool operators would be required to collude. In that case the pools users would learn of this and stop using that pool.



Collateral transactions from a successful DarkSend session are effectively destroyed using a sigScript to make them valid only for a given period of time.

**Improved Anonymity**

An anonymity enhancement to the generic CoinJoin implementation is added by only allowing inputs of the same size into the DarkSend pools. These sizes are referred to as “denominations” and are in powers of ten  (for example, 1DRK, 10DRK, 100DRK, 1000DRK). This allows the inputs from all users to be virtually the same. Outputs per user must add up to the denomination size.



Users that send less money than the denomination size will use a second “change” output. These outputs are new addresses not connected to their identity. This implementation allows for amounts of any precision to be sent without a negative impact in the quality of anonymity.

All users  entering a DarkSend transaction pool have an equal chance of becoming the master node. All participant nodes know which node is the current master by way of an election algorithm. Master nodes also have a collateral transaction that is made out to the payment node, which can be cashed if they misbehave in any way.

In the case where a master node loses internet connection or is a bad actor, the collateral transaction of that node will be cashed and a slave node will be elected in it’s place. Due to the trustless nature of DarkSend, there is no risk of lost money from the master node being a bad actor as a slave node would be elected to replace the master node and the collateral would be forfeited to the network.

**Master Node Election**

The election algorithm is a pseudo random deterministic algorithm based on the transaction IDs in the DarkSend pool. By adding up the hash values of the transaction IDs, and running the value through the X11 hashing algorithm, a pseudo random number is created.

Pseudo code:

```
Target = X11 (txid1+txid2+txid3+txid4) // txid = Transaction ID

NodeValue = X11(txid1+outputPubkey1+outputPubkey2)
NodeValue2 = X11(txid2+outputPubkey3+outputPubkey4)
NodeValue3 = 0 //last node to enter pool can’t be master

Score = Abs(Target-NodeValue)
Score2 = Abs(Target-NodeValue2)
```

This random number is compared to a target number derived from the txid and pubkeys of the users outputs. The node with the lowest score is elected master while the second lowest score is elected slave. By using this algorithm we achieve a decentralized tamper-proof system in which the users can know which node the master is.

**Master Node Responsibilities**

The decentralized nature of DarkSend requires that one node will decide which transactions are allowed into the pool to deal with network propagation issues. The master node is elected each round to broadcast the finalized transaction that will be signed by the DarkSend participants.

The participants will be able to check the authenticity of the messages coming from the master node by utilizing ECDSA signatures for all messages after election.

Participants in a DarkSend will only sign the finalized transaction if they find that their inputs and outputs are present with the correct amounts. After the transaction is signed and confirmed to be valid, the master node will broadcast the finalized signed transaction and resign.

**Improved Pool Anonymity**

Users who want to increase the anonymity of the pools can run scripts to “push” DarkSend transactions through the pool by sending money to themselves with DarkSend. This will allow them to take up a space in the pool to ensure the anonymity of other users. If enough users run scripts like this one, the speed of transactions and the anonymity of the network will be increased.

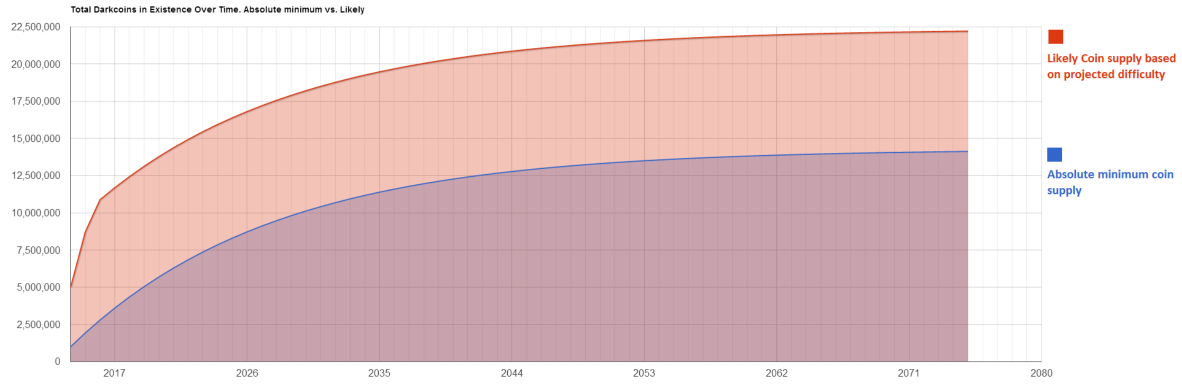
**Reward Curve vs Reward Halving**

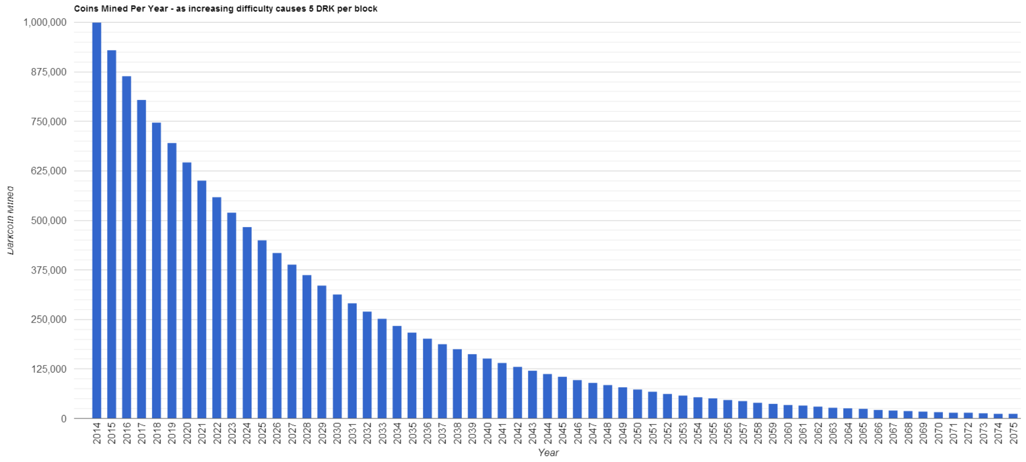
Bitcoin was designed to have a fixed supply with a declining block reward schedule. This makes Bitcoin a deflationary currency, with a money supply that grows a small percentage year-over-year. One problem with this approach is the abrupt reward halving that happens every four years. This could eventually cause large distortions in the mining network when the profitability of mining changes drastically overnight.

DarkCoin replaces abrupt reward halving with a reward curve, 2222222/(((Difficulty+2600)/9)^2). The maximum and minimum amounts are set to 25 and five respectively.

Using this formula, the reward will gradually drop over the following months and years - and then provide a steady supply of approximately one million coins per year. This is an inflationary model which strays from Bitcoin in a fundamental way. To create a fixed supply rewards will gradually taper off over time at a stable rate of 7% annually. This results in the projected total number of coins reaching approximately 22 million over the time frame shown - slightly more or less depending on how quickly hash power is added.

### DarkCoin Currency Supply and Mining Reward Schedule





## Difficulty Retargeting Using DarkGravityWave

DarkGravityWave uses multiple exponential moving averages and a simple moving average to smoothly adjust the difficulty. This implementation resolves possible exploits in KimotoGravityWell by limiting the difficulty retargeting to 3 times the 14 period EMA difficulty average.

## Proof-Of-Work Utilizing X11

Darkcoin uses a new chained hashing algorithm approach, with many new scientific hashing algorithms for the proof-of-work. X11 consists of blake, bmw, groestl, jh, keccak, skein, luffa, cubehash, shavite, simd, and echo.

Because it is more complicated than a SHA256 ASIC implementation, the use of X11 will prevent the use of ASIC miners for the short-term to mid-term future. It will also allow for a longer period of mining for CPU/GPU users.

GPU miners that mine with the X11 algorithm are currently experiencing reduced power usage (up to 50%) and reduced heat generation compared to scrypt.

## References

Joseph Bonneau, Arvind Narayanan, Andrew Miller, Jeremy Clark, Joshua Kroll (2014)
Mixcoin: Anonymity for Bitcoin with accountable mixes
http://eprint.iacr.org/2014/077.pdf

Fuh-Gwo Jeng, Tzer-Long Chen (2010): An ECC-Based Blind Signature Scheme
http://ojs.academypublisher.com/index.php/jnw/article/viewFile/0508921928/2053

Nakamoto S. (2008): Bitcoin: A peer-to-peer electronic cash system.
http://www.bitcoin.org/bitcoin.pdf