MimbleWimble

Lead Research Analyst
Mrinalini Bhutoria (Ria)
@riabhutoria

Research Intern
Wilson Withiam
@wilson_withiam

Cryptographer
Mira Belenkiy

Updated
March 07 2019

MimbleWimble is a privacy-enhancing and scalable blockchain protocol. It verifies that all transactions are valid without storing the blockchain's entire history. It was named after the tongue tying curse in Harry Potter that prevents the afflicted from spilling secrets. It was proposed in 2016 by someone under the pseudonym Tom Elvis Jedusor[1] who shared a Tor link on the bitcoin wizards IRC chatroom to a text file outlining MimbleWimble - then disappeared.

BACKGROUND

Andrew Poelstra, a mathematician at Blockstream, was intrigued by the protocol and published a more technically detailed and robust paper on MimbleWimble in October 2016. MimbleWimble is a blockchain protocol and Grin and Beam are its first two implementations. We explore MimbleWimble, Grin, and Beam in this report.

	Grin	Beam
Price ($)	$3.27	$0.68
Market Cap (2050)	$3,191,309,136	$178,005,938
Current Market Cap	$13,485,987	$5,258,784
24hr Volume ($)	$16,954,399	$6,076,770
Supply % Issued	0.4%	3.0%
% Down from ATH	42.6%	78.7%

Source: messari.io/onchainfx, coinmarketcap.com (as of 3/3/19)

Many people in the crypto community have been closely watching the MimbleWimble protocol because it aims to improve upon key problems with bitcoin and other cryptocurrencies in that, for the first time, it optimizes privacy *and* scalability.

- **Full privacy:** MimbleWimble hides the transaction sender, recipient, and amount from anyone not involved in the transaction. Observers see a transaction consists of some encrypted inputs and outputs. They can verify that the inputs are already on the chain, and the currency in the outputs sums to the same value. This is an improvement over systems like Bitcoin, where everyone can trace value as it gets transferred from one address to another.
- **Efficiency:** MimbleWimble lets validators store only unspent UTXOs. All other cryptocurrencies force miners and outside validators store the entire transaction history for the blockchain. This enables space savings and faster sync because as blockchain history grows larger, miners may be forced to use multiple drives to store the entire history.

[1] Tom Elvis Jedusor is the French name of Voldemort.

A validator checks a MimbleWimble transaction by verifying that (1) the sum of the inputs equals the sum of outputs, and that (2) transactions don't contain negative amounts to ensure that no transaction is attempting to mint new coins. The only transaction that can mint coins is the coinbase transaction, which is also the only identifiable transaction. However, validators and observers do not see who receives the block reward.

Another important distinction of MimbleWimble is that there are no addresses or public keys; there are only inputs and outputs. Each UTXO has a secret key, and the receiver stores the UTXO secret in his wallet. To send the UTXO, the sender must contact the receiver in a private channel and perform a multi-round communication to construct a transaction. The sender uses his UTXO secret to sign the UTXO, while the receiver gets a new secret for the output UTXO as a result of the communication.

THE PROBLEM

Blockchains are unforgeable public ledgers of transactions. Unforgeability means that users can send only funds that they received - they cannot send funds sent to others or create funds out of thin air. Bitcoin and similar blockchains publicize the sender address, receiver address, and transaction amount so that it is easy to verify that amount sent equals amount received and that the one sending the inputs is the one with the private key corresponding to those inputs.

The public nature of bitcoin (and other crypto assets) can be undesirable for people and businesses who don't want their transaction details to be shared with everyone. Additionally, with the rise of chain analysis firms like Elliptic and Chainalysis, researchers can tie outputs to illicit transactions and blacklist said outputs. The CEO of Binance once tweeted that they were able to freeze funds sent to the exchange by hackers after it was reported on social media. However, some argue that this goes against the principle of fungibility. Fungibility is the idea that all coins are created equal just as one dollar bill equals another dollar bill regardless of one bill's past activity.

Bitcoin and all other blockchains require miners and other validators to store the entire transaction history of the chain to verify that all transactions are valid. New participants need to download and verify the entire blockchain to validate the current state of the network. This places significant space, time and computation requirements on new participants who want to sync with the network. The size of the Bitcoin blockchain right before 2019 was about **200GB.**

MIMBLEWIMBLE'S SOLUTION

MimbleWimble uses cryptography in a clever way to achieve unforgeability while, at the same time, optimizing privacy and scalability. Rather than verifying each output's entire history as in Bitcoin, a MimbleWimble implementation checks whether the sum of all inputs minus the sum of all outputs on the blockchain is zero to validate the chain (this reinforces one of the fundamental features of money where amount sent equals amount received). MimbleWimble uses a combination of confidential transactions, CoinJoin, rangeproofs, and cut-through to do so.

Similar to Bitcoin, MimbleWimble relies on elliptic curve cryptography and the UTXO model. However, MimbleWimble is a much more stripped down version and trades off programmability due to the privacy downsides of script. As a result, more complex and elaborate functions like time-lock or payment channels like Lightning Network cannot be executed (as it stands).

CRASH COURSE: UTXOS

Bitcoin and MimbleWimble use an unspent transaction output (UTXO) model to account for balances. This model can be compared to using coins and bills to pay for goods and services versus using a credit or debit card. For example, Alice wants to buy a $30 shirt but she has two $20 bills. She can't just give the merchant one and a half bills. Rather, she gives the merchant both bills and receives a $10 bill back as change.

The UTXO model functions in a similar way: Alice has two transaction outputs of 1 BTC and 0.5 BTC from prior transactions. Alice needs to pay a merchant 1.3 BTC. Her wallet creates a transaction which sends 1.5 BTC with two new outputs. The merchant receives 1.3 BTC and Alice receives 0.2 BTC back as change (less transaction fees). Bitcoin users can check the block explorer and will notice that their bitcoin address often sends a higher amount of bitcoin than specified.

CRASH COURSE: ELLIPTIC CURVE CRYPTOGRAPHY

Elliptic curves have several properties that make them useful for complex cryptographic protocols. One property of elliptic curves is one-way functions. It is easy to take a random point G on an elliptic curve and multiply it by some integer s to get another point P=sG. However, given (P,G) it is **computationally infeasible** to recover the value of s. This lets us use (P,G) as a public key, and s as a secret key. Another property of elliptic curves is that points on an elliptic curve have useful algebraic properties:

1. **Distributive: (a+b)G = aG+bG**
2. **Commutative: a(bG) = b(aG) = (ab)G**

CRASH COURSE: PEDERSEN COMMITMENTS

Pedersen commitments are cryptographic primitives that combine the one-way and algebraic properties of elliptic curves. A commitment to some values (x,y) is computed as P=xG+yH. While computing s=P/G is computationally infeasible, computing (x,y) from (P,G,H) is impossible because there are countless combinations of x and y that would satisfy the relationship P=xG+yH. However, a user who knows a single pair (x,y) that satisfies this relationship cannot compute a second pair (x',y') that satisfies this relationship without violating the one-way property of elliptic curves.

CONFIDENTIAL TRANSACTIONS

MimbleWimble relies on a cryptographic concept called Confidential Transactions (CTs) to achieve privacy by default. Confidential transactions were proposed by Gregory Maxwell, who drew his inspiration from Adam Back's homomorphic encryption for Bitcoin. Confidential transactions use Pedersen Commitments to hide the value of a UTXO.

In MimbleWimble, a transaction output or input **is represented as** a Pedersen Commitment rG + vH. G and H are random points on an elliptic curve and are public parameters of the blockchain. The value v is the UTXO value and r is the blinding factor and functions as the secret key for the UTXO. The value rG is the corresponding public key. MimbleWimble uses Pedersen commitments to obfuscate sensitive transaction information instead of showing plaintext transaction values. Pedersen commitments permit the use of basic arithmetic to validate transactions.

Consider an example where we have two inputs and one output. We provide sample values and blinding factor, while leaving the public parameters G and H as variables.

	Value (v)	Blinding factor (r)	rG+vH (Pedersen Commitment)
Input 1	1	5	5G+H
Input 2	2	6	6G+2H
Output	3	11	11G+3H
Output-Input			(11G+3H) -(5G+H)-(6G+2H)=0

By verifying that output commitments minus input commitments equal zero, we can confirm that no new money was created without knowing the actual input and output values. This works only if the values of the inputs sum to the value of the output and the blinding factor of the inputs sum to the blinding factor of the output.

TRANSACTION KERNEL

The problem with confidential transactions as outlined above is that they require the input and output UTXO to use the same blinding factor, which is the recipient's secret key. If the sender learns the value of the recipient's blinding factor, she can steal the recipient's output UTXO. MimbleWimble overcomes this problem using zero-knowledge proofs.

Consider a simple example of sending 5 coins. The sender has an unspent UTXO represented by the commitment X=45G+5H, where 5 is the value and 45 is her blinding factor (r), or secret key. The recipient chooses a random blinding factor 7 and creates an output UTXO represented by the commitment Y=7G+5H. A verifier that compares inputs to outputs will see the commitment of the excess:

X-Y = (45G+5H) - (7G+5H) = 38G

MimbleWimble calls the value 38 the **excess or kernel**, and the value X-Y = 38G the **transaction kernel**. In a valid transaction, the transaction kernel is always of the form X-Y = rG+0H, where r is some integer. This is true even if multiple inputs and outputs are used, If the sum of the input values is equal to the sum of the output values, the value multiplied by H will be zero. A valid transaction kernel is always in the form of a public key. The sender and receiver each know part of the corresponding secret key. MimbleWimble has a protocol which lets them jointly compute a signature using their blinding factors to sign the transaction. The kernel represents a multisig key for transaction participants.

RANGEPROOFS

The MimbleWimble protocol requires the transaction amounts to be positive so users cannot create coins from thin air. As we mentioned, the only transaction type that can mint coins is the coinbase transaction. Rangeproofs are a cryptographic technique used to prove that, given a Pedersen Commitment X, the prover knows a pair of integers (r, min < v < max) such that X=rG+vH. MimbleWimble implementations use rangeproofs to prove that v is greater than zero. MimbleWimble uses recently introduced **Bulletproofs**, a type of rangeproof that only consumes **~5kb to ~700 bytes**.

NO ADDRESSES

As we mentioned, MimbleWimble does not use addresses. A key motivation behind removing addresses is to enhance privacy and address bloat. In MimbleWimble-based blockchains, users must communicate off-chain to create a transaction. The sender shares a proof that she controls some coins with a recipient who accepts control of those coins. Because there are no addresses that publicly assign control of coins, there is no "standard" way to send transactions. As a result, transaction participants need to set up a chat session to share the proof of control and transfer control to the recipient. This is very different from bitcoin (and most other blockchains), where transactions can be executed without participation from the recipient.

COINJOIN

One way to combat the public nature of transactions is CoinJoin. CoinJoin is a way to combine inputs into a single large transaction that makes it is difficult to distinguish which inputs are paying which outputs. CoinJoin has been implemented in JoinMarket, ShufflePuff, DarkWallet, SharedCoin, Wasabi, Samourai. The downside of wallet-based CoinJoin is that users have to opt-in to use the service. This diminishes its effectiveness because users either aren't aware of these services or don't care enough to go through the trouble of using them, resulting in a small set of CoinJoined transactions (a small "anonymity set"). This does not effectively hide originating addresses and destinations. Additionally, users must interact to create CoinJoin transactions since every input owner must sign the entire combined transaction to authenticate it.

In MimbleWimble, users don't need to opt in CoinJoin is enabled by default. A block no longer has individual transactions. Rather, it looks like one large transaction. Figure 1 is a simplified version of an untouched set of transactions to be included in the next block. In Figure 2, MimbleWimble joins the transactions together in a process similar to CoinJoin so that what is left is a single transaction that has combined a list of all inputs and a list of all outputs.

Figure 1

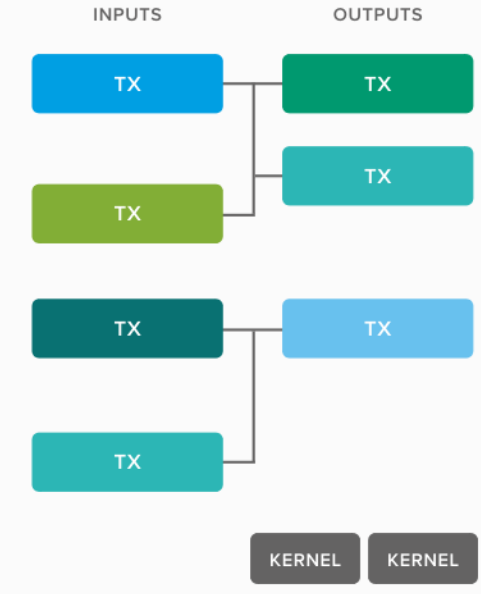
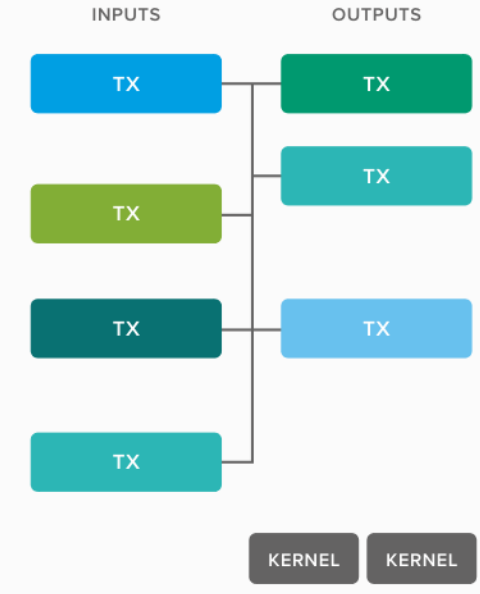


Figure 2



CUT-THROUGH

Confidential transactions are much larger than regular transactions. CTs are a little less than five times larger than non-CTs. MimbleWimble addresses this challenge using a technique called cut-through to provide efficiency improvements.

As we mentioned above, downloading the full Bitcoin blockchain consumes almost 200GB of space and is growing. If all the transactions on Bitcoin were confidential transactions like MimbleWimble, the size of the blockchain would be orders of magnitude bigger.

MimbleWimble uses a process called “cut-through” to remove redundant outputs that are used again as inputs within the same block to free up space within a block and reduce the amount of data that needs to be stored on the blockchain, while maintaining the same level of security. This is best illustrated through a diagram, adapted from Andrew Poelstra’s presentations.

In Figure 3, the network identifies that the teal output is used as a later input. The network removes this redundancy from the input/output in this given block to slim down the data that must be stored. While MimbleWimble removes the evidence that cancels out, it maintains the authorization that these transactions took place, aka the kernel.

Figure 3

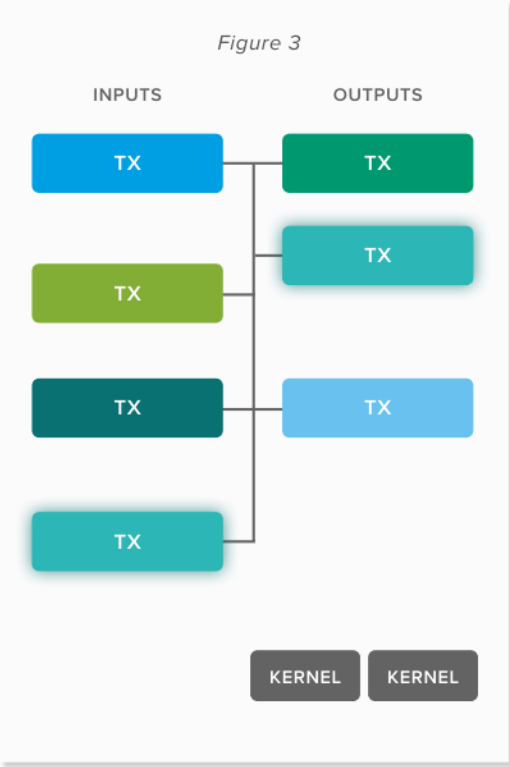
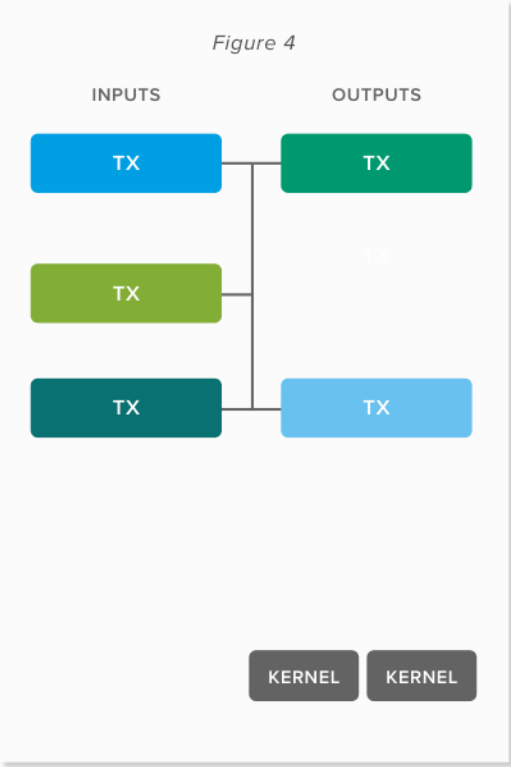


Figure 4



Not only does MimbleWimble use this tool without blocks at the micro level, it employs cut-through at the macro level so that **the only pieces of data that remain are the block headers, UTXOs and transaction kernels.** Nodes can use these key pieces of data to validate the blockchain. This means that the blockchain could shrink if, for example, there's a block that spends more outputs than it creates. Theoretically, this could reduce the amount of data needed to prove that the state of the ledger is correct over time.

Grin, the first implementation of MimbleWimble, **has shared** that the size of a given MimbleWimble node will be "on the order of a few gigabytes for a Bitcoin-sized blockchain, and potentially optimizable to a few hundreds of megabytes." **According to Grin,** assuming 10 million transactions with 100K UTXOs the size of a ledger without cut-through will be about 130GB with 128GB of transaction data, 1 GB of transaction proof data, and 250MB of block headers. With cut-through, the size of the blockchain could drop to 1.8GB, with 1GB of output data, 520MB of UTXOs, and 250MBs of block headers. Beam believes its blockchain size could be 30% of Bitcoin's when it reaches the same scale.

DANDELION

The biggest threat to MimbleWimble privacy is that nodes could record the transactions as they are broadcast to the network, before they are included in a block. Before cut-through, transaction outputs spend some time in the mempool (unconfirmed transaction pool). This allows spy nodes to build transaction graphs[2] and possibly discover the sender IP.

Dandelion, which is not part of MimbleWimble, is a supplemental feature that Grin and Beam incorporate. Dandelion tries to lower the chance of spy nodes successfully creating a transaction graph by "delaying the appearance of the transaction on the network by relaying it quietly first before bursting it out" (**Andreas Antonopoulos**).

Usually, when someone sends a transaction to the blockchain, it is broadcasted to all nodes on the network. Dandelion divides the broadcasting of a transaction into two phases, starting with the "stem" or "anonymity" phase where it is broadcasted randomly to one node, which randomly sends it to another, and so on, until it reaches the "fluff phase" where the system broadcasts the transaction to the entire network. This prevents a watching node from mapping a transaction using Dandelion back to the originating address. Dandelion++ provides improvements over Dandelion that make it even more difficult to create transaction graphs.

In MimbleWimble, transactions can also be merged before the stem phase to make it even more difficult to link inputs to transactions. Beam takes this one step further and makes it possible to add dummy or decoy outputs in situations where there aren't enough outputs to merge.

A key challenge with Dandelion was that, during the stem phase, if the transaction is passed to a node that subsequently goes offline, the transaction would not be propagated to the network. Grin and Beam address this challenge - if this a transaction doesn't reach the fluff phase within a reasonable amount of time the transaction is automatically broadcasted to the broader network.

[2] Transaction graph is a technical way of describing all the inputs and outputs of all transactions for money in and money out.

SCRIPTLESS SCRIPTS

MimbleWimble does not support transaction scripts, which are an important feature of most blockchains. Script is code embedded in transactions that enables basic smart contract

functions. Without it, MimbleWimble cannot support conditional transactions, use time-locks, state channels (e.g. Lightning network), cross chain atomic swaps, and so on. This is the price of unlinkable transactions and cut-through. Verifiers have no way of checking if smart contract conditions were met because the relevant UTXOs and their conditions might have been deleted.

When the first MimbleWimble paper was published in August 2016, unconditional transactions seemed to be a limitation the community would need to live with. However, Andrew Poelstra found a way to implement simple smart contracts off-chain using **scriptless scripts**. Scriptless scripts are based on the idea that blockchain validators only need to check that signatures are present and correct. They don't need to know the conditional elements that take place off chain. Schnorr signatures can be used to support scriptless scripts.

Specifically, participants of a transaction can create a Schnorr *multi-signature* by combining their individual signing keys to interactively produce a signature, which is the only piece of data that needs to be submitted to and validated by nodes.

Aaron Van Wirdum provides a **great explainer**, which we adapt here. He uses the example of a streamer who wants to listen to an artist's song. The artist and streamer need to submit their combined Schnorr signature to the blockchain to validate the conditional transaction. The artist, who has the rights to the song has a secret song key, 7000. The artist's half of the Schnorr signature is 8000. The artist can create an "adaptor signature" of 1000 by subtracting the secret song key (7000) from her piece of the Schnorr (8000). The artist then shares the adaptor signature with the streamer who uses cryptographic tricks to confirm that it equals the artist's piece minus the secret key. The streamer then shares her piece of the Schnorr signature with the artist. Let's say it's 5000. The artist submits the combined signature (8000+5000=13000) to the blockchain, automatically revealing her signature (8000) to the streamer. The streamer can now back into the secret song key (8000-1000=7000) to listen to the song.

This all happens off-chain such that no one besides the artist and streamer ever discovers the individual values and steps. The only thing validators see is the combined Schnorr signature of 13000. Adaptor signatures are undetectable by the public. Nothing other than the "settlement transaction" is recorded on the blockchain. Scriptless scripts can be implemented by adding a new opcode that enables Schnorr signatures. Grin and Beam are in the process of implementing scriptless scripts and there isn't an exact timeline around when the functionality will go live.

Scriptless scripts have the potential to enable things like confidential assets, cross-chain atomic swaps, and second layer scaling solutions like Lightning on MimbleWimble blockchains. Scriptless scripts are not required to be implemented on MimbleWimble, and might even spread to other blockchains.

CONCLUSION

MimbleWimble is based on proven cryptographic primitives. Some of the building blocks have been published in peer-reviewed cryptographic journals, while others are public whitepapers and technical reports. The first MimbleWimble blockchains Grin and Beam have only recently launched. While MimbleWimble is a new and experimental technology, it has the potential to provide significant privacy and scalability benefits, but it has unsolved UX and privacy challenges. Much testing and iteration is needed to make privacy work on open blockchains

at scale. At the moment, concepts that seem to be impenetrable could face unexpected issues in practice.

On the UX front, there are no addresses and transacting parties need to interact and be online (though not necessarily at the same time) to sign and complete transactions. This is not intuitive when compared to existing crypto assets.

On the privacy front, miners can see transactions in the mempool before CoinJoin and cut-through take place. As a result, nodes watching the network can build detailed transaction graphs, compromising privacy, which is opposed to MimbleWimble's key value proposition. Though there are potential solutions like Dandelion and dummy UTXOs, they have yet to be perfected in practice.

OTHER PRIVACY SOLUTIONS

MimbleWimble is not the first or only approach to blockchain privacy. While a fully comprehensive and thoughtful discussion of all privacy solutions available is out of the scope of this report, it is important to touch upon the alternatives. These include (though are not limited to) other protocol or base layer privacy coins (Zcash, Monero), second layer privacy solutions (Blockstream side chains), and transaction layer privacy (via wallets like Samourai and Wasabi).

PRIVACY COINS

Two privacy coins that launched before Grin and Beam are Zcash and Monero. These coins implement privacy at the protocol layer. Monero is a privacy coin based on the CryptoNote protocol. A key advantage in Monero is that privacy is on by default. It hides sending and receiving addresses and transactions. Monero achieves privacy using Ring Confidential Transactions and stealth addresses. Ring signatures add "decoys" to transactions without exposing which coins were really signed, effectively mixing the coins. Monero's main downside is that transactions are ten times as large as Bitcoin transactions, even with the implementation of Bulletproofs, that offers significant space savings.

Zcash is based on the Zerocash protocol design. Zcash uses shielded addresses to hide transacting parties and zk-snarks (a type of zero-knowledge proof) to hide transaction amounts. Unlike Monero (and MimbleWimble-based Grin and Beam) privacy is not on by default in Zcash. Before Zcash's Sapling update, creating a confidential transaction was computationally heavy and time consuming, deterring users from opting in. With the Sapling update, the memory and time required for shielded transactions has gone down, which could encourage the use of shielded transactions. The other downside of optional privacy is that choosing to shield transactions could be seen as dubious. Another criticism is Zcash's trusted setup. While Zooko Wilcox **has said** breaking the trusted setup won't compromise privacy, Peter Todd (a Bitcoin researcher) **disagrees** based on conversations with a zk-snarks developer.

SIDECHAIN

A sidechain is an independent blockchain that is linked to a base layer protocol via a two-way peg. The two way peg enables coins from the original chain to be interchanged with sidechain assets at a fixed rate following a verification process. These supplemental chains

can support alternative features and consensus mechanisms beyond the capabilities of base layers to optimize for solutions including, but not limited to, privacy and scalability. Bitcoin sidechain company Blockstream has deployed one such network, the recently launched Liquid, that incorporates confidential transactions by default. Liquid uses a small group of 15 known members (called functionaries) to validate transactions and produce blocks, which accelerates transaction times at the cost of decentralization. While Liquid governance is more concentrated, it addresses particular issues seen with exchanges such as the ability to redeem LBTC, Liquid's native token, on any functionary within the Federation. This would be especially useful if a single network member was down. Moreover, Liquid's design disincentivizes a functionary from controlling escrowed bitcoin without also comprising their reputation and the network. One slight criticism of Liquid is a few of the its trusted intermediaries consist of unregulated and historically insecure cryptocurrency exchanges such as Bitfinex and OKCoin.

PRIVACY WALLETS

The advantage of wallet based privacy solutions like Wasabi, Samourai or Breeze is that they can be implemented on top of bitcoin (or other coins) without changing the underlying protocol. Criticisms include small anonymity sets and transaction delays if matching funds aren't immediately found. For example, Samourai's Staggered Ricochet can take up to two hours before reaching the final destination. Also, wallets are privy to the rules of their centralized platforms. At the beginning of 2019, **Google made Samourai remove** certain privacy and security features from its app as it violated the Google Play store's new rules.

Despite the plethora of options for enhancing privacy, these are all early stage technologies (including MimbleWimble, Grin and Beam). Each have their own trade-offs and, at this point, there is no clear answer to the best approach to privacy in crypto.

Grin

Grin is the first open-source implementation of MimbleWimble in the programming language, Rust. Grin documentation was released on October 20, 2016 by an anonymous developer by the name Ignotus Peverell. Many core Grin developers have similarly adopted Harry Potter related monikers. Grin released four testnets before launching on mainnet on January 15, 2019. Grin was and is praised by the crypto community for its similarities to bitcoin - notably, its anonymous development team, its fair launch (no premine, ICO, or founders reward) and its donation-based funding model. However, Grin does have several notable differences.

- **Monetary policy**: Grin is designed to be used as a medium of exchange rather than as store of value like bitcoin.The miner reward in Grin is 60 grins per one minute block (or 1 Grin per second), perpetually. This equates to high inflation in the early days that trends down over time.
- **Consensus algorithm**: In the beginning, Grin will try to achieve decentralization by using two proof-of-work algorithms that are variants of Cuckoo Cycle (one is ASIC friendly and the other is believed to be ASIC resistant because it is memory-hard). Cuckoo Cycle is a new and somewhat controversial proof-of-work algorithm; the Handshake blockchain whitepaper describes some concerns.
- **Governance**: Grin does not have a formal governance process but does have a technical council comprised of eight members that manage Grin's general fund and development roadmap. It also holds open governance and development meetings.
- **Features**: Grin is working to augment the MimbleWimble protocol by adding features such as scriptless scripts that allow for more complex and conditional transactions. Members of the community are also working to improve the user experience through solutions like grinbox and wallet713.
- **Challenges:** While Grin is celebrated for its donation based funding model, it is also a challenge to rely on external donations to continue building and improving. Additionally, there is much work ahead to make Grin usable for non-technical users.

Grin has been listed on multiple exchanges since launch, though it does not solicit exchange listings or pay listing fees. While the community is happy to help exchanges list the token, **Ignotus Peverell said** they do not "worry too much about externalities and [things they] don't really have control over.

MINING ALGORITHM

Initially, the Grin team was planning to use two algorithms that were believed to be ASIC resistant due to their memory bound requirements: Cuckoo Cycle[3] (developed by John Tromp in 2015) and an Equihash algorithm called Equigrin with a high memory requirement.

Memory requirements were thought to limit calculations to CPUs and high range GPUs. When Cuckoo Cycle was developed, it was believed to be ASIC resistant due to its SRAM (static random access memory) **requirements**. Algorithms bound by SRAM were thought to make manufacturing ASICs more difficult and expensive (**Source**). John Tromp, the creator of Cuckoo Cycle, said that "originally Cuckoo Cycle was designed to make memory latency a bottleneck. Now, many years later, we realize that the SRAM that Cuckoo Cycle makes excellent use of [...] is quite affordable in ASICs. We expect ASICs to have a large efficiency advantage over GPUs." John Tromp explains Cuckoo Cycle in depth in this **podcast episode**.

In August 2018, the community acknowledged **that** ASICs are (1) inevitable in practice[4] and (2) can be detrimental to bootstrapping a distributed community in the beginning but are not necessarily bad in the long run. On the contrary, an ASIC friendly algorithm can help make a network more secure as ASIC machines increase the hashrate of the network, making it more difficult and expensive to attack. ASICs can be good for the long term success of a protocol, because miners who have invested tens of millions are aligned with the protocol in terms of security. Also, according to **Derek Hsue**, "any attempts to generate sustained ASIC-resistance will result in secret ASICs—which are problematic."

As a result of the above points, Grin then decided to switch to proof-of-work algorithms that are both variants of Cuckoo Cycle, the primary ASIC friendly (AF) algorithm and the secondary ASIC resistant (AR) algorithm, and phase out the second algorithm. The primary algorithm in Grin is called Cuckatoo31+, a more AF version of Cuckoo Cycle. It is AF in that it uses **hundreds of MB of SRAM** to provide efficiency gains over GPUs. The secondary algorithm, called Cuckaroo29, is a memory-hard AR PoW algorithm. However, the only way to truly guarantee ASIC resistance is to undergo planned hard forks that consistently adjust the algorithm (a la Monero), rendering ASICs built for it obsolete. Grin will execute such forks every six months to adjust the algorithm to disincentivize the production of ASICs for this algorithm until it is phased out at the end of two years.

Some members of the crypto community **are concerned about the stability of the Cuckoo Cycle algorithm**. John Tromp first proposed it in 2014, and in its short time it has gone through several revisions as researchers found ways to optimize computation. Cuckoo Cycle is based on a graph problem. One concern is that a miner may gain an advantage if it figures out how to compute Cuckoo Cycles faster than the rest of the network. According to John Tromp, its relative advantage might increase with migration to larger graphs. This advantage would go away if the rest of the community implemented the same solution.

Starting off, Grin has been structured such that 90% of blocks are mined with the secondary algorithm and 10% of blocks are mined with the primary algorithm. At the end of two years, 100% of blocks will be mined with the primary algorithm. Over the next two years, Cuckatoo31+ will receive a greater proportion of the block rewards, linearly increasing by 3.75% a month. The Grin community hopes that there will be healthy competition from multiple ASIC manufacturers by the time Cuckatoo31+ makes up 100% of mining. Grin retargets difficulty every block based on a sixty block window.

[3] Cuckoo Cycle searches very large graphs for a cyclic path rather than hashing so the solver speed is best measured as graph searches per second. Cuckoo Cycle consumes far less energy than most other proof-of-work algorithms and is memory bound rather than compute bound. Yeastplume explains it well in this podcast episode: say we draw circles on a piece of paper and draw lines between all of them at random. The algorithm figures out if you can find a path or cycle between these nodes such that if I start at one particular circle and follow the random lines along a path to get back to the same circle or node.

[4] "Around 2014 when ASIC miners were first commercialized, the tactic of using memory-hard algorithms to resist ASICs made sense due to the higher cost of RAM. In the past few years, the sharp decline of RAM cost makes it possible for ASIC designers to manufacture machines for these networks at increasingly low cost. Memory-hard algorithms cannot keep ASICs at bay indefinitely." Leo Zhang

GRIN MINING POOLS

According to miningpoolstats.com there are fifteen pools mining on Cuckaroo29 and eleven mining pools devoted to Cuckatoo31+. At the time of writing, the combined hashrate of the top two pools (Sparkpool and F2pool) is 82% of the total hashrate on **Cuckaroo29** and 68% on **Cuckatoo31+**. Both Sparkpool and F2pool have provided donations to Grin's developer fund and general fund. While it appears that hashpower is concentrated among mining pools, mining pools are comprised of many participants who can choose to leave the pool and direct their mining power elsewhere as they please.

The third largest pool is GrinMint, a mining pool **launched by BlockCypher** first as a testnet in September 2018 and on mainnet in January 2019. BlockCypher collects 2.5% in fees and has said that it will allocate 0.5% to the Grin developer community. BlockCypher also has a developer devoted full-time to Grin (Quentin Le Sceller). Other pools that give back to the Grin community include MimbleWimble Grin Pool and grin-pool.org.

One of the criticisms against Grin was that venture capital backed miners controlled significant hashrate when Grin launched. As a consequence, market participants who would have been buyers turned into sellers of the cryptocurrency. As mining pools discover blocks and receive mining rewards, they have to immediately sell the coins because they pay miners in bitcoin.

MONETARY POLICY

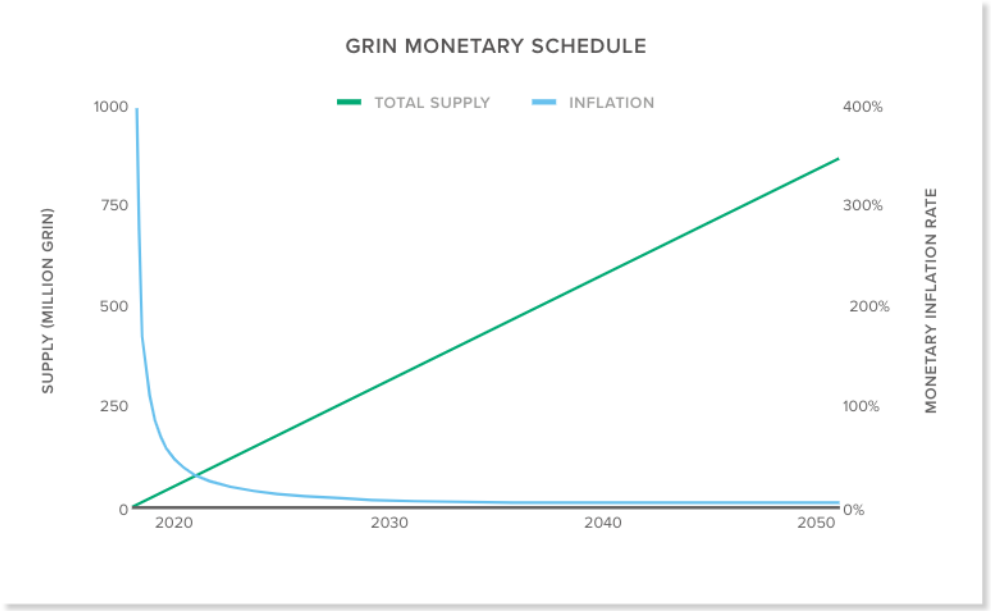
Grin has linear emission rate and will be released at a rate of 60 Grin per one minute block (one Grin per second) forever - its supply is intentionally uncapped. Bitcoin, on the other hand, has a hard cap of 21 million and has deflationary supply schedule as the block reward is halved every four years until it reaches near zero around the year 2140. This makes bitcoin valuable as a store of value as the model encourages holding coins due to the expectation that the value per coin will increase over time.

In Grin, the rate of inflation will be very high in the early days and approach zero over time when there are millions of coins in circulation, though it will never reach zero. In practice, it will take 10 years before inflation falls below 10%, and 25 years before it falls to 4% (about the same rate as bitcoin in 2018). It will take 50 years for inflation to fall below 2%. However, in reality, the Grin team believes inflation will be lower when accounting for lost coins. **According to the team**, lost coins can be as high 2% of total supply per year, and should be excluded when calculating inflation rates. Perpetual issuance was seen as a potential solution to mitigate the effect of lost coins.

Another reason behind perpetual inflation is that (1) an inflationary policy rewards and favors participants more equally than does a disinflationary policy, regardless of whether they joined the network in the early days or not (it is not a "get-rich quick scheme for early miners"), and (2) if the value of a coin is expected to be similar tomorrow relative to today, then it has a greater chance of being used as a medium of exchange, which is Grin's intention. High inflation in the short to medium term incentivizes spending rather than storing, because coins experience significant dilution. The community also believes that the incentive to spend can potentially enable wider distribution of coins.

Also, perpetual emission could prevent Grin from eventually having to rely solely on a fee market to secure the network -- a challenge the Bitcoin community is discussing/facing right now. Once bitcoin's issuance reaches near zero, the network will have to transition to

transaction fees alone to reward miners for their work in securing the chain. This is a new economic model for blockchains, and many questions remain: how many transactions will be required per block, what will be the minimum fee per transaction to secure the network, and how will challenges presented by layer two solutions that aim to reduce fees impact the security of the underlying blockchain.



Source: https://www.grin-forum.org/t/emmission-rate-of-grin/171/21

Skeptics have criticized Grin for it's uncapped, linear emission rate because inflation reduces the purchasing power of savings, which negates an asset's use as a store of value. However, the inflation in Grin is an intentional design choice to encourage spending, preempt the issue of lost coins, and ensure the network always has a way to compensate miners for securing the network. One downside of high inflation is that block rewards are currently a significant portion of total supply, though this is similar to the early days of bitcoin. This can negatively impact the value of the coin as mining pools sell Grin rewards for bitcoin to pay their miners.

GOVERNANCE

Grin's Lehnberg says, "Governance is about how decisions are made that are seen as legitimate [sic] Legitimate, in the eyes of the participants (those involved in making decisions) and the stakeholders (those affected by those decisions)." As it stands, Grin does not have an explicit governance process though is transparent about outstanding debates and is open to community participation.

Grin has a technical council that manages the Grin General Fund and directs the development of the project. The members of the council **include** Ignotus Peverell, Antioch Peverell, Hashmap, Jaspervdm, Lehnberg, Quentin le Sceller (BlockCypher), Yeastplume (Michael Cordner), John Tromp and Gary Yu. Anyone can participate in governance and development meetings and discussions, but the most active contributors generally play a greater role.

The technical council holds bi-weekly governance and development meetings where topics can range from ASIC resistance, raising and directing funds, major flaws and bugs, security audits, exchange integrations, hard forks and so on. Grin also publishes an agenda and notes and transcripts from the meetings on **the Github page** before and after they take place. There is also a **Governance section** on grin-forum, which has consistent posts on the topic, indicating the community is actively thinking about how to approach governance in the long run.

The technical council-led governance and development process has allowed the community to be quick and nimble in the early days, to avoid slowing down the progress of the network. However, there has been **discussion** around establishing a more structured governance process with checks and balances as Grin grows and matures. Council members and contributors have articulated that it will be necessary to implement a more formal process that establishes:

- A more structured way for the community to communicate feedback on governance and development topics.
- Rules around the scope of the council's authority and a way for the community to provide input on council members.
- The opportunity for all stakeholders to make their voice heard. Stakeholders include core developers, one-off contributors, miners, users, investors, exchanges, and so on.

The downside is that the council adds an element of centralization, and in the long run, an unofficial council can be dangerous. An illustrative example is the Burst PoCC, which served a similar function as Grin's technical council. One day, they became upset with the community and unexpectedly quit, but kept access to the repository, DNS registration, and more. They also took additional malicious actions such as cheating mining pools and selling Burst early, that ultimately hurt the Burst blockchain.

FUNDING

Grin is an open source project that is entirely donation-based. While it is lauded for it's fair launch - no ICO, premine, or founder's rewards, the downside is that development and progress are slow. Grin relies on unpaid part-time volunteers and collecting donations for the core developer fund, security audits, marketing and web development, conferences, and **more**.

As **Tushar Jain** points out, "without the capitalist incentives, development will be delayed." This is a truth the Grin community recognizes. On the general fund page, they say, "the reality is that a project of Grin's scope would be greatly helped by having a source of funding. This would allow Grin to reach its potential more quickly, more reliably, with better supporting infrastructure, and with a far greater chance of success competing against (or co-existing with) well-funded blockchain projects."

The Grin community started building Grin in 2016 and only launched on mainnet in January 2019. Meanwhile, Beam, another implementation of MimbleWimble (discussed in further detail below), is a private company backed by VC investors that began working on the project at the beginning of 2018 and launched a week before Grin.

Additionally, Yeastplume (Michael Cordner), a core developer and instrumental member of the community initially had a hard time raising funds that would allow him to devote his full attention to Grin. **Only after** Ignotus Peverell **expressed his disappointment** that **Cordner's**

campaign (€55,000) was far from being even 10% funded did the campaign start to see an uptick in donations. It has since exceeded its goal, raising €66,580 at the time of writing. For a full list of donators, see this Friends of Grin **list**.

Relying on donations may work in the short run. However, to sustain development and attract talent to the network, Grin will have to rethink its funding model as it faces increasing competition from well-funded projects with paid employees. In this well articulated thread on **developer incentives**, Nathaniel Whittemore proposes an alternative model where "for-profit businesses that sit atop the open protocol 1) provide sufficient incentives to attract top talent, while 2) continuing to contribute to core development roadmaps."

USER EXPERIENCE

As we explained above, MimbleWimble does away with addresses. As a result, sender and receiver must relay messages (called "transaction slates") off-chain to interactively negotiate the transfer of coins. There are multiple ways to relay the standardized JSON messages. One method is file based transfer, where the file contains the JSON message in plain text and can be transferred in a number of ways (email, Telegram, Keybase, HAM radio, carrier pigeon, etc.), and another method is the URL method, where an API accepts raw JSON in text format.

A group of third-party developers under the name vault713 is working to make Grin more usable and widely adopted. Their first project is a transaction protocol called Grinbox. Grinbox is a message relaying service that simplifies the transaction process when used with wallet713, which is a fork of the core Grin wallet by vault713 that currently runs on Linux. Grinbox and wallet713 aim to improve the process of sending and storing Grin.

As a first step, they allow participants to create public addresses to send/receive funds so that they don't have to expose their IP address. wallet713 also allows users to link contact names to addresses stored locally on their computers. Additionally, wallet713 allows for asynchronous transaction building. vault713 is also working on adding more features that enhance privacy and usability **such as** multi-sig support, atomic swaps between BTC and Grin, CoinJoin with other wallet713 users before transactions enter the unconfirmed transaction pool, mobile/web/desktop GUI and more.

More options for creating transactions will emerge as the protocol matures and attracts more talent. This could include close proximity technologies based on NFC, QR, Bluetooth, etc. Eventually, the market will likely coalesce on a convenient and well understood solution, but it will take some time to get there and it remains to be seen what method become the standard.

Grin is only a few months old and, as it stands, the protocol is best suited for technically savvy users that put in the time and effort to understand how it works. While the community is starting to address user experience challenges through efforts like grinbox and wallet713, it will take time, iteration, and education before non-technical users feel comfortable transacting on the network.

CONCLUSION

Grin is a project that originally drew the attention of cypherpunks and crypto-anarchists, but the similarity of Grin's ethos to bitcoin attracted the attention of many. Namely, Grin has been praised for its anonymous leader, its donation-based and grassroots funding model, its focus on privacy and decentralization, and its community that cares deeply about advancing the project rather than making a quick buck.

But launching Grin on mainnet was just the first step. There is much work to be done to set up Grin for long-term success and widespread adoption. Key challenges that need to be addressed include a more reliable way to raise funds, a more intuitive user experience to draw more users to the network, and research to address privacy holes in the system (i.e. the ability for watching nodes to create transaction graphs).

The core team has said that its "primary focus remains stability, performance, and security. Nurturing a healthy ecosystem with third party development teams integrating Grin into their services and products is also crucial for adoption to improve." This need not come from core Grin developers. Rather, these challenges can be addressed as a third party developer ecosystem emerges around the Grin blockchain.

Grin is still a very new project that pioneers new and untested ideas, cryptographic concepts, and technologies. If Grin can address key challenges, it has the potential to emerge as a way to put privacy back in the hands of individuals.

Beam

Beam is a VC-backed startup based in Israel that launched a privacy-focused cryptocurrency of the same name based on the MimbleWimble protocol on January 3, 2019. It began building its implementation in C++ in March 2018 and launched a testnet in September 2018. While Beam and Grin are similar in that they are both implementations of MimbleWimble aiming to provide privacy enhancements to their users, they differ in their approach. Unlike Grin, Beam is a private company that employs developers to work on the implementation. Beam started off closed-source but later opened up its development. Another important differentiator in Beam is optional auditability, targeted at businesses and regulators.

- **Monetary policy**: The supply schedule for Beam is deflationary with the block reward dropping 50% after the first year and subsequent halvings occuring every four years until a hard cap of ~263 million beam is reached. Also a 20% dev tax will be placed on block rewards, paid out to the Beam Treasury, to help fund future Beam development.
- **Mining algorithm**: Beam uses a modified version of Equihash, a proof-of-work mining algorithm, to provide network consensus. To ensure decentralization, Beam will remain ASIC-resistant for the first 12-18 months by regularly adjusting its algorithm.
- **Governance**: Beam currently operates as a VC-backed startup with paid employees. The long-term goal is to pass full governance on to a non-profit foundation that manages the Beam Treasury and maintains the protocol.
- **Features**: Beam is adding an auditability feature so businesses can demonstrate compliance and provide visibility of transactions without compromising their privacy. Beam developers are also exploring a secure BBS system that will enable non-interactive offline transactions.
- **Challenges**: Bootstrapping a PoW protocol is an arduous task, and avoiding ASIC mining integration will keep the overall hashrate low, which correlates with the cost to attack the network. Moreover, Beam's current operational and governance structure is centralized, and a shift towards a more decentralized model will require avoiding a power struggle between all invested parties.

MINING ALGORITHM

Beam uses **Equihash**, which is a proof-of-work algorithm created by Alex Biryukov and Dmitry Khovratovich at the University of Luxembourg. Equihash is an asymmetric, memory bound algorithm based on the **generalized birthday problem**. Another key property of Equihash is that mining is **stochastic**, meaning the likelihood of generating a proof is independent of past successes or failures. Equihash has two parameters that can be adjusted: n (width in bits) and k

(length), which determine the complexity of the underlying problem and thus the memory and time complexity of the algorithm. Beam is launching with Equihash parameters of n=150 and k=5.

Equihash is asymmetric in the sense that it requires a lot of memory to generate a proof, but it does not require a lot of memory to verify it. This is an important property of Equihash as most other memory-bound algorithms are not asymmetric i.e. it is just as difficult to verify a proof as it is to generate it. Memory bound refers to the fact that time taken to generate a proof scales with memory rather than processing power. Equihash imposes disproportionately higher computational requirements if less memory is used.

Initially, memory was an expensive resource, so ASICs were not assumed to provide significant memory optimizations over regular CPUs and GPUs. On the other hand, ASICs provide significant bandwidth improvements over GPUs, which provide significant bandwidth improvements over CPUs. Due to infrastructure improvements, the cost of optimizing ASICs for memory is no longer as high as expected.

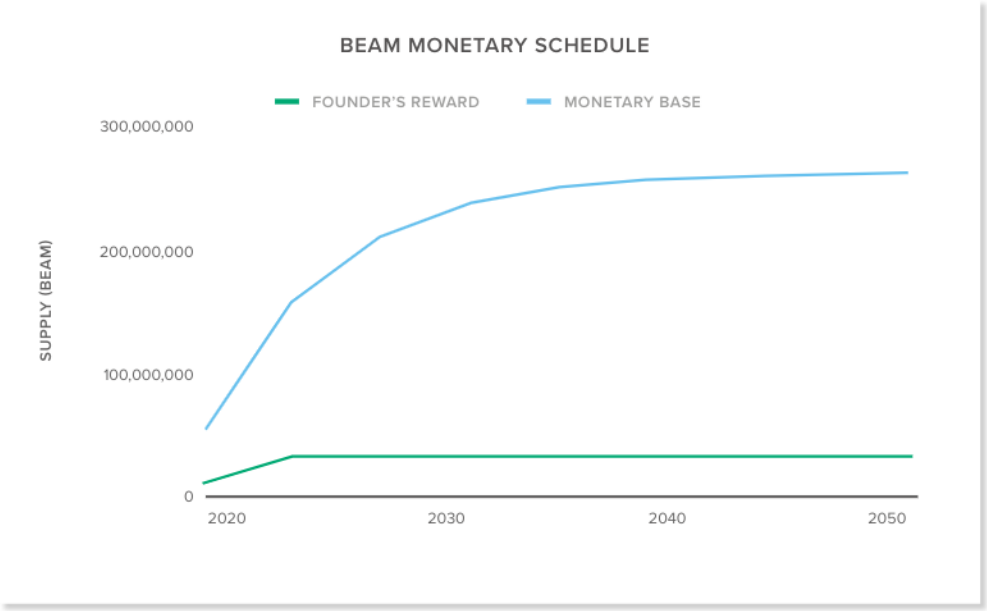
Zcash, a privacy-focused crypto asset that also uses Equihash, initially chose Equihash because it was believed to be ASIC resistant. However, in 2018, Bitmain released the Antminer Z9 mini to mine Zcash more efficiently than commodity hardware "**by interfacing** with SRAM at a relatively low cost". In Beam's post on Equihash, they highlight that "researchers at the University of Luxembourg discovered that as of May 2018, 20%-30% of Equihash is mined by ASICs."

Beam says it has set its Equihash parameters to give CPU and GPU miners an advantage over ASICs in the short run so that the initial distribution of coins is more widespread. However, it recognizes that ASICs are inevitable and even desired in the long run as they have a sunk-cost investment and increase the network hashrate, in turn making it more secure and more difficult to attack.

MONETARY POLICY & FUNDING

Beam has a monetary policy that resembles that of Bitcoin. It features a hard cap and deflationary emission schedule that undergoes a regular block reward halving (a 50% drop in the amount of coins earned for each block mined) until the inflation rate reaches zero. Therefore, the startup expects Beam to be used as a store of value as opposed to a medium of exchange like Grin. The Bitcoin similarities end there, however, as Beam includes a higher block reward during its first year, a Founders Reward on coinbase payments for the first five years, and one minute block times.

In year one, the block reward will be 100 beam, a higher than usual reward in order to incentivize miners to join the network early on and introduce Beam to the market. A 20% (founder's fee/devtax) is programmed for the first five years, so of the 100 beam mined per block in year one, 80 beam will be paid to miners and 20 beam will be paid to the Beam Treasury. For years two through five, the block reward will drop 50% to 50 beam, with 40 beam paid to miners and 10 beam paid to the treasury. In year six, the block reward will drop 50% again to 25 beam (all of which will be paid to miners) and going forward, halving will occur every 4 years until year 129. Block rewards will stop at year 133, at which point Beam expects to have a total capped supply of ~263 million beam.



Source: https://medium.com/beam-mw/mimblewimble-emission-schedule-215551948259

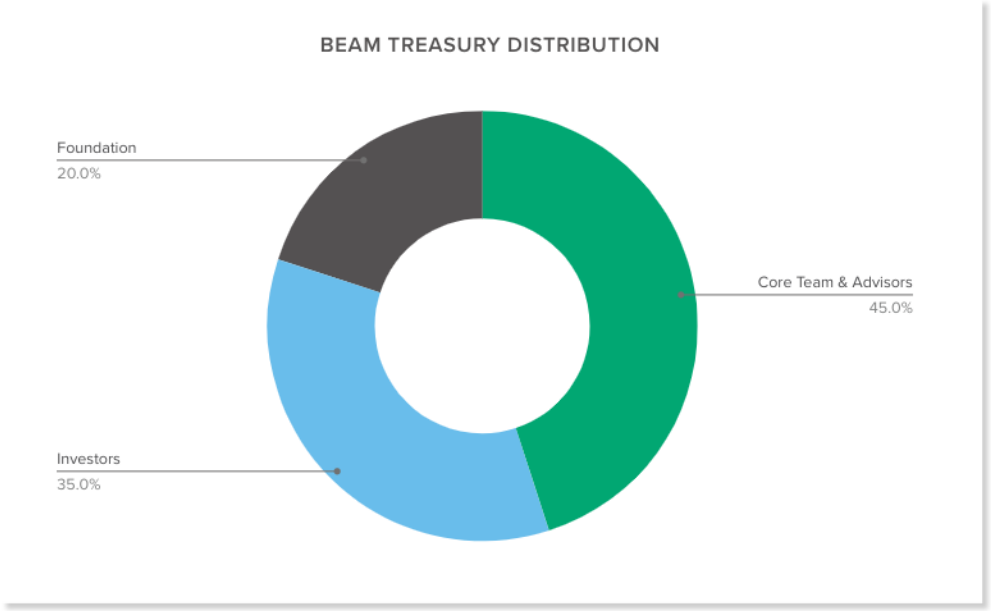
Beam adopted a Founders Reward, also referred to as a Dev Tax, in order to refund investors and provide financial motivation for ongoing protocol and tooling development. A founders reward or fee is supplemental code built into a blockchain protocol that automatically splits and routes the block reward (the coinbase transaction) between the miner of the block and the founding team's known address.

This approach is decidedly different from a pre-mine, like an ICO, or an insta-mine, as seen with Dash, which compensate crypto asset founders with a large, liquid lump sum. While both are desirable to early team members, these compensation designs often lack effective treasury management or vesting schedules. As a result, the misappropriation of funds and exit scams are prevalent in systems that promote instant gratification.

The Founder's Reward was designed to instead compensate founders gradually as the project is developed. Initial stakeholders are therefore better incentivized to align interests and preserve the long-term success of the network. Moreover, the reward system is baked into the protocol, providing inherent fund allocation transparency and the freedom to "exit with low friction via [a] soft or hard fork," as indicated by **Arjun Balaji**.

This founder's incentive structure was initially designed and popularized by the Electric Coin Company (formerly known as Zcash Company), the venture behind the development and maintenance of the privacy-focused cryptoasset Zcash. At first, Zcash miners will only earn 80% of the block reward. The remaining 20% will be distributed amongst the Zcash foundation (an independent nonprofit that supports Zcash development), the Electric Coin Company, and early Zcash developers and advisors. After these first four years, the Founder's Reward is pre-programmed to drop to zero ensuring 100% of newly minted Zcash will go to miners until its cap of 21 million is reached.

The Beam funding model echoes that of Zcash with a 20% founder's fee during its early stages paid to the Beam Treasury. Unlike Zcash, Beam's version will carry out for its first five years, including the first year when the block reward is 100 Beam. At the end of those five years, a cumulative amount of over 31.5 million Beam should be routed to the the Beam Treasury. The plan is to allocate 35% of the funds to repay early investors and another 45% of the funds to core team members and advisors in periodic installments. The remaining 20% will be used to support the Beam Sovereign Money Foundation, which is the project's long term solution for protocol maintenance and governance.



Source: https://medium.com/beam-mw/mimblewimble-emission-schedule-215551948259

In addition to the Founder's Reward, Beam has raised a minimum of $5 million from various VC funds, including Recruit Co. LTD, Yeoman's Capital, and Node Capital, to employ full-time developers dedicated to advancing the protocol. These investors will be paid back in regular Beam installments as part of the Founder's Reward, aligning the interests of each stakeholder.

The core Beam team and early investors both recognized a more concentrated effort would speed up production and avoid the lengthy update or pivot delays often seen in community run projects. Therefore, Beam stakeholders opted for this more centralized approach to guide the project through its infancy. As Beam continues to mature, the goal is to achieve a more decentralized incentive and governance structure, handing block rewards to network miners and control to the community.

On the downside, criticisms against Beam include that its launch did not give all investors equal accessibility. Crypto assets that raise money from investors prior to launch on mainnet or dedicate a portion of funds to exclusive groups (i.e. ICOs, Founder's Rewards, or premines) could result in an unequal distribution of coins.

Standing in opposition are launches like those exhibited with Bitcoin and Grin, where the crypto asset can only be earned through traditional PoW mining. Technical hurdles aside, any interested investor could join the network and mine new bitcoin or Grin. Such launches tend to exhibit a more equal distribution of funds amongst network users.

GOVERNANCE

In its current state, Beam relies by a small VC backed team based in Tel Aviv to determine any protocol updates and added features. Therefore, the organizational structure of the project resembles that of a private startup more than the governance processes exhibited by most decentralized protocols. This gives Beam the ability to control risk factors and the freedom to quickly pivot and iterate to meet market demand and accelerate productivity in its early stages.

The Beam leadership team is comprised of CEO Alexander Zaidelson, CTO Alex Romanov, COO Amir Aaronson, and CMO Beni Issembert. Additional core members consist primarily of developers as well as a few designers and department directors. The company has also enlisted the insight from a group of twelve advisors, including Maja Vujinovic (CEO of OGroup and former CIO of Emerging Tech at GE) and Marco Streng (CEO and Co-Founder of Genesis Mining).

As the protocol matures, the founding members will look to shift control outside of the original team to the Beam Sovereign Money Foundation, an independent non-profit foundation intended to be run by prominent and respected community members. Beam believes establishing the foundation will help achieve its goal of a decentralized organizational structure. The process of defining the foundation's responsibilities and rules and attracting its board of directors is set to take place over the next several months with an expected launch before the end of 2019. Once the foundation starts to gain traction, the current Beam company plans to transition into a service provider role that builds end-user applications on top of the Beam protocol.

Most of the information regarding the foundation establishment process has yet to be released, but what is known about the role of the Beam Foundation includes:

- Managing the proposals and development of improvements to the Beam protocol
- Funding and promoting research related to Beam, MimbleWimble, and Dandelion
- Increasing awareness to help grow the community
- Pushing the importance of privacy within digital currency and financial sovereignty

CHALLENGES

By adopting a startup model, Beam is subject to the typical issues associated with most startups and will face further difficulties trying to assuage public perception and shift to a more decentralized governance model. Startups in general suffer from a high failure rate due to a number of factors, including lack of product/market fit, high burn rates leading to insufficient funds, and internal team conflict. A team of experienced entrepreneurs and advisors is far from a guarantee for long-term success, and a single internal conflict could threaten the entire project.

The more daunting task is earning enough support to help move protocol governance and development work from the small original team to the community as a whole. An important cryptoasset evaluation metric is the level of decentralization associated with a project, which

Beam has intentionally chosen to delay. The argument in favor of Beam's strategy is early stage projects "**need the freedom to be able to quickly pivot and iterate**." In the words of **Arjun Balaji**: "It's next to impossible to simultaneously build new types of distributed networks while optimizing for decentralization early on" as these objectives are inherently at odds.

USER EXPERIENCE

BEAM WALLET

Beam launched with a **graphical user interface (GUI) wallet** for non-technical users as well as a command line interface (CLI) wallet for Mac, Windows, and Linux. The Beam desktop wallet creates public addresses that transacting parties can share with one another. These addresses are not recorded on the blockchain. Beam also recently launched a beta version of an Android mobile wallet and plans to roll out an iOS mobile wallet. It also says it is in talks with hardware wallet providers to roll out support for Beam on hardware wallets.

SBBS

Beam tries to make offline transaction creation and asynchronous communication more seamless and secure by using a Secure Bulletin Board System (SBBS). Beam's BBS is designed after bulletin board systems that were popular in the eighties and early nineties. People with home computers and a modem could connect to other computers using a landline and leave messages for others to see on text-based bulletin board systems (BBSes). BBS hosts were people who **converted their computers** into a digital meeting ground. As they grew more advanced, users could play text-based games and even facilitate file transfers.

In Beam, BBS wallets are comparable to home computers plus modems (they are the "client") and Beam full nodes are comparable to BBS hosts (they serve the purpose of the server). The SBBS is part of the node software and is maintained off-chain. BBS full nodes create a store-and-forward network to relay messages to recipients who are offline. Messages are encrypted using public keys and are then relayed to receiving wallets via Beam full nodes. In this case, public keys act as addresses in the P2P system. If the receiving wallet is offline, store-and-forward Beam nodes can store messages in a database that acts as a message board. Participants try to decrypt messages on the message boards they are subscribed to, but only the participant with the corresponding private key can decrypt the message directed to them.

Beam intends to use it's wallet and SBBS to make the user experience **similar** to transacting with address-based blockchains and lower the barrier to entry to interacting with a crypto asset based on the novel MimbleWimble protocol.

BEAM WALLET CHALLENGES

Shortly after launch (January 9), Beam developers discovered a vulnerability in their wallet that would allow user funds to be compromised. Developers found that they left something in the wallet code that shouldn't have been there. While Beam underwent multiple code reviews and audits prior to launch, they were mainly focused on the robustness of Beam's cryptography implementation, indicating that the same amount of rigor might not have gone into auditing the wallet and SBBS. Beam announced that the patch was discovered and fixed internally and that no funds were stolen. Users were advised to uninstall the wallet and re-download an updated version from the Beam website.

On January 21, Beam experienced another issue that caused the blockchain to temporarily halt and stop producing blocks at block 25,709. The reason was improper wallet usage. More specifically, the same UTXO was sent in separate transactions to the blockchain by cloned wallets. This led to "incorrect cut-through processing and ultimately to an invalid block." Beam did not produce blocks for almost three hours and did not process transactions for about five hours.

AUDITABILITY

One of Beam's key differentiators is its focus on serving business. In addition to the improvements made possible by MimbleWimble, Beam is also developing an opt-in compliance and auditability feature (Wallet Audit or business wallet) to help businesses comply with regulations and perform required audits. This allows businesses to create a wallet that has auditor keys attached to it so auditors can identify tagged transactions on the blockchain created by the business wallet. With this optional compliance feature, transactions will still be private but users will be able to report them to auditors if they need to. This opens up the use cases of crypto assets to regular businesses.

According to Zaidelson, while the actual information will be generated by the wallet and stored off-chain, the blockchain will store references to the information per transaction as hashes. The Beam blockchain does not store historical transaction details - it only stores transaction kernels referring to past transactions. In **this interview,** Zaidelson says Beam "can use this kernel to store additional [encoded] information...including compressed hashes of documents like invoices or receipts." When users undergo an audit, the auditor can check that the data matches the cryptographic hash of the data.

This functionality is still a work in progress, which creates some uncertainty around how well it will work in practice. If it works, however, it could solve a major pain point for businesses who must currently choose between crypto assets that fall at very opposite ends of the spectrum. On the one hand, crypto assets like bitcoin provide full transparency and auditability at the cost of disclosing confidential information to competitors. On the other hand, using the privacy features in crypto assets like Zcash and Monero hides all traces of transactions, prohibiting any kind of auditability.

The challenge with auditability is that businesses have to store the data corresponding to the hash in a secure way off chain. Additionally, businesses need to trust that auditors won't share the auditor keys with other parties that are not authorized to see the data. While Beam could create a way to share private data, auditors might not have the technical know how to conduct audits using tagged transactions on Beam's blockchain. In theory, they could outsource this function, but this would expand the group of people with access to sensitive data.

ROADMAP

Shortly after launching on mainnet, Beam published a comprehensive roadmap for 2019. It is divided into two key categories, Beam Core (focused on improving and advancing the core Beam protocol) and Beam Compliance (focused on launching and iterating on Beam's compliance and auditability initiatives for businesses). Longer term, Beam has spoken about an initiative called Project Lumini, which will focus on creating a bridge between Beam and some other smart contract blockchain(s) and rolling out confidential assets on Beam.

BEAM CORE

Beam Core is divided into five phases - Agile Atom, Bright Boson, Clear Cathode, Double Doppler, and Eager Electron. The highlights from the roadmap include implementing the Lightning network as a second layer solution on Beam for faster payments by the end of the year, rolling out beam to bitcoin atomic swaps by the end of March 2019, and undergoing two planned hard forks to adjust the Equihash mining algorithm for initial ASIC resistance, among other initiatives detailed below. We note that Beam first has to roll out smart contract and multisig functionality (via scriptless scripts, for example) to to support layer two solutions like Lightning Network.

Phase	Timeline	Details
Agile Atom (AA)	February 2019	Mainnet release Beam Lightning Network position paper
Bright Boson (BB)	March 2019	Beam to bitcoin atomic swaps Hardware wallet support Mobile wallet for Android Lightning PoC on Beam testnet Payment processor integration
Clear Cathode (CC)	2Q19	First planned fork to modify Equihash Mobile wallet for iOS Multisig support Lightning Alpha on Beam mainnet
Double Doppler (DD)	3Q19	Research alternative consensus mechanisms Porting Beam to languages like Rust Enhance wallet security Lightning Beta on Beam mainnet
Eager Electron (EE)	4Q19	Second & last planned fork to modify Equihash I2P/Tor integration to improve privacy Alternative consensus PoC based on research in DD

BEAM COMPLIANCE

The primary objective of the Beam Compliance track is to make Beam usable by businesses. Beam plans to include a “Compliant Wallet” and a “Regulator interface” in its compliance suite that is expected to be tailored for country-specific regulations. As of now, the tentative go-live date is 2020.

Timeline	Details
1Q19	Partner with “design partners” from traditional finance and crypto industries Complete and publish go-to-market plan
2Q19	Compliant Wallet” PoC Share PoC with pre-alpha users for feedback
3Q19	“Compliant Wallet” Alpha “Regulator interface” Alpha Start approaching regulators
4Q19	“Compliant Wallet” Alpha 2 “Regulator interface” Alpha 2 Customer trials

CONCLUSION

Beam takes a commercial approach to building a store-of-value privacy coin. It is VC-backed and has paid employees that devote their full attention to the project. As a result, Beam has been able to go from development to launch in less than a year. It has a clear focus on user experience and ease of use through its work on the Beam wallet and secure messaging system. On the other hand, it has already experienced some hiccups with the desktop wallet that could have led to lost funds, which could have been detrimental to such a young project.

Beam has outlined large plans in its 2019 roadmap, including Lightning Network on Beam and auditability solutions for businesses and regulators. Beam is unique in its choice to build-in optionality for business users who currently have to choose between blockchain platforms that offer extreme transparency or extreme privacy. However, Beam’s compliance and auditability solutions are not yet live and could open up additional attack vectors. Beam has ambitious goals that should be thoroughly tested before they are rolled out on mainnet to avoid careless mistakes that could subject users to compromised funds or data. If Beam can deliver on its plans, it could offer a unique set of features that solve clear problems for business users.

Recap

MimbleWimble is novel in that it offers privacy and efficiency enhancements hand in hand through a unique combination of the protocol's version of confidential transactions, CoinJoin, and cut-through within and among blocks to make it possible for more devices to partake in securing network.

Grin and Beam are both implementations of MimbleWimble, but their similarities stop there. Ignotus Peverell (Grin's creator) **points out** that "a common misconception is that MimbleWimble describes a full cryptocurrency solution and therefore tend to lump Beam and [Grin] in the same basket."

While both projects seek to offer their users privacy and efficiency improvements, they differ in most technical, structural, and organizational elements. The difference that has ignited the most discussion is the sustainability of Grin's donation and volunteer-driven/cypherpunk approach (similar to Bitcoin and Monero) versus Beam's VC-backed startup approach with it's founder's reward and paid employees (similar to Zcash). Time will tell which approach is superior. Until then, it will be interesting to see how these projects play off of and learn from one another.

Category	Grin	Beam
Launch Date	January 15, 2019	January 3, 2019
Technology		
Consensus	PoW - Cuckoo Cycle	PoW - Modified Equihash
Blocktime	~1 min	~1 min
Speed	17 tps	**17 tps**
Monetary Policy		
Use Case	Medium of Exchange	Store of Value
Emission	Constant emission rate, disinflationary	Deflationary
Max Supply	None	~263 million
Governance	Technical Council	Beam Development Limited & Beam Sovereign Money Foundation
Funding	Donations to Grin general fund	VC and Founders Fee
Community		
Contributors	116 (8 primary)	12 (4 primary)
Commits	1900+	4400+
Twitter	13,000+	8900
Privacy enhancing features	Dandelion	Dandelion Decoy outputs
Roadmap	Scriptless scripts Confidential assets Atomic swaps	Mobile wallet (Android, iOS) Scriptless scripts Confidential assets Atomic swaps Compliant wallet Lightning network

DISCLOSURES

This report has been prepared solely for informative purposes and should not be the basis for making investment decisions or be construed as a recommendation to engage in investment transactions or be taken to suggest an investment strategy in respect of any financial instruments or the issuers thereof. This report has not been prepared in accordance with the legal requirements designed to promote the independence of investment research and is not subject to any prohibition on dealing ahead of the dissemination of investment research under the Market Abuse Regulation (EU) No 596/2014. Reports issued by Circle Internet Financial Limited ("Circle") or its affiliates are not related to the provision of advisory services regarding investment, tax, legal, financial, accounting, consulting or any other related services and are not recommendations to buy, sell, or hold any asset. The information contained in this report is based on sources considered to be reliable, but not guaranteed, to be accurate or complete. Any opinions or estimates expressed herein reflect a judgment made as of the date of publication, and are subject to change without notice. Trading and investing in digital assets involves significant risks including price volatility and illiquidity and may not be suitable for all investors. Circle and its affiliates trade and hold positions in digital assets and may now or in the future trade or hold a position in the asset(s) that is the subject of this report. Moreover, Circle and its affiliates may now or in the future provide financial or other support related to the asset(s) that is the subject of this report. In addition, employees and other associated persons may trade and hold positions now or in the future in the asset(s) that is the subject of this report. As a result, Circle, its affiliates, and its employees or other associated persons may now or in the future receive remuneration with respect to the asset(s) that is the subject of this report and may have certain conflicts of interest in connection with the provision of this report. Circle will not be liable whatsoever for any direct or consequential loss arising from the use of this information.

MimbleWimble sources

https://medium.com/beam-mw/mimblewimble-explained-like-youre-12-d779a5bb483d

https://github.com/mimblewimble/grin/blob/master/doc/intro.md#balance

https://bitcoinmagazine.com/articles/confidential-transactions-how-hiding-transaction-amounts-increases-bitcoin-privacy-1464892525/

https://medium.com/@jcliff/behind-mimblewimble-cd9da78a00e9

https://bitcoinmagazine.com/articles/coinjoin-combining-bitcoin-transactions-to-obfuscate-trails-and-increase-privacy-1465235087/

https://uncommoncore.co/grin-and-the-mythical-fair-launch/

https://www.datadriveninvestor.com/2019/02/09/what-is-mimblewimble-and-its-implementation-in-grin-beam/

https://github.com/mimblewimble/grin/blob/master/doc/intro.md

https://github.com/mimblewimble/grin/blob/master/doc/grin4bitcoiners.md

https://bitcoinmagazine.com/articles/battle-privacycoins-why-monero-hard-beat-and-hard-scale/

https://www.youtube.com/watch?v=SojKwudx1TI

https://soundcloud.com/mindtomatter/ltb-356-privacy-on-the

https://thecontrol.co/an-overview-of-privacy-in-cryptocurrencies-893dc078d0d7

https://en.bitcoin.it/wiki/CoinJoin

https://blockonomi.com/dandelion-protocol/

https://medium.com/@jcliff/behind-mimblewimble-cd9da78a00e9

https://medium.com/@hendi.richter/cryptocurrencies-mining-pools-and-grassroots-community-funding-7a6d8f1768e9

https://www.theblockcrypto.com/2019/01/08/mimblewimble-history-technology-and-the-mining-industry/

https://cyber.stanford.edu/sites/g/files/sbiybj9936/f/andrewpoelstra.pdf

https://medium.com/blockchain-capital-blog/crypto-innovation-spotlight-2-scriptless-scripts-306c4eb6b3a8

https://www.coindesk.com/grin-and-beam-a-tale-of-two-coins-being-built-on-mimblewimble

https://www.tokendaily.co/blog/blockchain-privacy-equal-parts-theory-and-theater

https://bitcoinmagazine.com/articles/scriptless-scripts-how-bitcoin-can-support-smart-contracts-without-smart-contracts/

https://zcoin.io/zcoins-privacy-technology-compares-competition/

Grin sources

https://cryptonews.com/exclusives/interest-in-grin-exceeding-our-wildest-beliefs-says-team-3362.htm

http://diyhpl.us/wiki/transcripts/realworldcrypto/2018/mimblewimble-and-scriptless-scripts/

https://www.youtube.com/watch?v=3pd6xHjLbhs&feature=youtu.be&t=1h35m58s

https://twitter.com/nlw/status/1091319888562532355

https://www.tokendaily.co/blog/on-grin-mimblewimble-and-monetary-policy

https://tlu.tarilabs.com/protocols/grin-design-choice-criticisms/MainReport.html

https://github.com/mimblewimble/docs/wiki/Monetary-Policy

https://soundcloud.com/captain-crypto/pirates-parley-grin-michael-cordner-yeastplume

https://github.com/mimblewimble/grin/blob/master/doc/pow/pow.md

https://github.com/mimblewimble/docs/wiki/GPU-Mining-Stats

https://uncommoncore.co/grin-and-the-mythical-fair-launch/

https://knowtechie.com/qa-with-layer1s-alexander-liegl-how-grin-could-be-the-most-important-crypto-launch-since-ethereum/

https://blog.blockcypher.com/an-introduction-to-grin-proof-of-work-103aaa9f66ce

https://blog.blockcypher.com/mimblewimble-grin-a51e2b47ea35

https://coin360.com/blog/Grin-Token--Part-1--MimbleWimble-s-Origins--Audience--Criticism-And-Competition

https://coin360.com/blog/Grin-Token--Part-2--Anonymity--Scalability--Mining-And-Inflation

https://medium.com/blockchain-capital-blog/grin-governance-a-novel-approach-154aca07291b

https://soundcloud.com/mindtomatter/ltb-356-privacy-on-the

https://medium.com/iterative-capital/inside-grins-novel-approach-to-resisting-asics-4a99c088078a

https://github.com/mimblewimble/docs/wiki/GPU-Mining-Stats

https://moneromonitor.com/episodes/2017-09-26-Episode-014.html

https://minergate.com/blog/grincoin-and-mimblewimble-described/

https://twitter.com/TradeToken/status/1096391463884255237

https://www.coindesk.com/grin-launch-crypto-interest-from-deep-pocketed-investors

https://grin-tech.org/general_funding

https://breakermag.com/grin-founder-ignotus-peverell-on-life-after-launch-and-the-path-forward/

Beam sources

https://www.beam.mw/faq/what-is-the-difference-between-beam-and-grin

https://tlu.tarilabs.com/protocols/grin-beam-comparison/MainReport.html

http://btcnewspaper.com/what-is-the-beam-coin-mimblewimble-grin-vs-beam/

https://breakermag.com/beam-wants-to-split-the-cryptocurrency-atom-private-for-everyone-auditable-if-you-want-it/

https://www.mycryptopedia.com/equihash-algorithm-explained/

https://medium.com/beam-mw/speaking-of-equihash-1ab467d01f68

https://www.cryptolux.org/images/b/b9/Equihash.pdf

https://medium.com/beam-mw/mimblewimble-beam-roadmap-2019-b2c7f38fc106

https://medium.com/beam-mw/the-secure-bulletin-board-system-sbbs-implementation-in-beam-a01b91c0e919

https://medium.com/beam-mw/mimblewimble-wallet-vulnerability-report-7b8e88dc7e

https://medium.com/beam-mw/team-beam-discovered-a-critical-vulnerability-in-beam-wallet-658f2c0344e2

https://medium.com/beam-mw/mimblewimble-emission-schedule-215551948259

https://docs.google.com/spreadsheets/d/1vixXEPgmbS7C3BzJ10OVHPBJr_z2KpWf9G77ujNI5eQ/edit#gid=0

https://medium.com/@arjunblj/zcash-the-founder-incentive-trilemma-fe7689fc8293

https://z.cash/blog/funding/

https://twitter.com/ColliderVC/status/1098156478018539520

https://www.crunchbase.com/organization/beam-0b5a#section-locked-charts

https://www.recruit.co.jp/en/newsroom/2019/0218_18297.html

https://medium.com/beam-mw/mimblewimble-beam-roadmap-2019-b2c7f38fc106

https://medium.com/@arjunblj/centralize-then-decentralize-6de1a0f9d2b0

https://medium.com/@arjunblj/zcash-the-founder-incentive-trilemma-fe7689fc8293

https://twitter.com/vcorem/status/1100878186081075201

https://www.cbinsights.com/research/startup-failure-reasons-top/

Additional resources

MimbleWimble

Original MimbleWimble Whitepaper

Andrew Poelstra - MimbleWimble

Grin

Website

Twitter

Reddit

Github

Telegram

Discord

Community Forum

Messari resource page

Block Explorer (GrinExplorer)

Block Explorer (Blockscan)

Beam

Website

Beam Position Paper

Twitter

Reddit

Github

Telegram

Medium

Block Explorer