TABLE OF CONTENTS

About Us

We are an exceptional community driven project based on trust and transparency supporting decentralization. Our main goal is to empower people to restore their financial freedom.

What is 5G-CASH

5G-CASH(VGC) is a hybrid project with a growing vision to become a payment gateway for the community with no inflation keeping low transaction fees. It is also built on the basic protocol of Bitcoin and incorporates multiple improvements and additional technologies to make it perfectly transparent and anonymous at the same time, secure, economical, and efficient for all users. Our network is based on trust along with the features we provide being a hybrid network.

Why is 5G-CASH a Hybrid Network?

Because it uses **PoW** but also incentivizes holders to mint their balances to protect the network against 51% attack with the new version of **POSv3.0** to earn the same reward amount and once holders allocate the necessary coins they can run **Masternodes** to easily support the consensus, validating and securing the Network.

What makes 5G-CASH different from another currency?

Apart from its fair launch VGC has a fair chance of mining for everyone even with older GPUs. One of the new coming changes to the network is that miners will have the chance to mine VGC with a new hybrid algo in which **CPUs, GPUs, Android devices and even FPGAs or ASICs** will mine all together without the need to compete in hashes speed among them, being the quantities of mining devices the only reason one specific miner get more hash. VGC also prevents hungry miners from attacking the network making it hard to find a block every 5 minutes having **POSv3** in the middle.

What does VGC mean?

VGC is the fifth generation digital cash blockchain

V= Number 5 in roman numeral

G= Generation

C= Cash

Chain specifications

Name: 5G-CASH
Ticker: VGC
Type: PoW, PoSv3, Masternodes
Algorithm: X16Rv2
Block Time: 5 minutes
Mined Block Maturity: 20 blocks
Max Supply: 100,000,000 VGC
Masternode Collateral: 50,000 VGC

Mining rewards

Block reward : 3 VGC
Masternode reward: 45%
(PoS) and (PoW) share 55%

Ports

P2P port: 22020
RPC port: 22019

What Is Premining?

Premining is the act of creation of a quantity of blockchain-based tokens or "coins" before a cryptocurrency is launched to the public. Premining is associated with initial coin offerings (ICOs) as a way to reward founders, developers, or early investors into the project.

Did 5G-CASH have an ICO?

There was no ICO, just a fair launch in which everyone was alerted of its launch and gathered to mine it.

Why did 5G-CASH premine?

As our project is based on trust and transparency we would like to state that our first launch code was a disaster due to a bad consensus from developer @Barrystyle ----> https://github.com/barrystyle which made us look bad but we decided to dump his code and move forward. Along with the new code we respected people's effort, time and money for all their spends to support the chain. That's why we premine 13,000,000 VGC to give people's coins back from the old chain.

VGC Chain Features

Sigma Protocol
Elysium Protocol
TOR Protocol
Dandelion++ Protocol
Proof of Work (PoW) X16R-V2
Proof-of-Stake (POSv3)
Masternodes
Dark Gravity Wave

Sigma Protocol

Sigma protocol involves a 3-round proof with the following

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

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

03 Proof provided by the prover to show that he actually knows and understands the truth without showing the Verifier how he did it

The **Sigma protocol** is a mechanism for proving that a statement or occurrence is true. It usually involves two participants; **THE PROVER** and **THE VERIFIER**. The prover's aim is to show that the statement or occurrence is really true without showing the verifier the key to understanding the statement or occurrence.

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

PROOF AND SAFETY SIZES

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

TRUSTED CONFIGURATION

Since the beginning of Zcoin, we have always seen the problem of "trusted configuration" as a major drawback. In a trusted configuration, some secret (public) parameters are generated based on a **"primary private key"**. These network parameters are needed to create so-called "**zero-knowledge proofs**", which is the **anonymity technology we use**. The **"primary private key"**, sometimes called **toxic waste**, needs to be destroyed. If this data is not destroyed, someone with access to that key can generate an infinite amount of anonymous coins. One of the main criticisms of Zerocash and zkSNARKs (which should not be confused with Zerocoin as used in Zcoin), as implemented in Zcash, is its requirement for having a reliable and controversial configuration. An easy way to view a trusted configuration is to create a box with a lock on it and its corresponding key. Owning the key will allow you to create unlimited treasure from the box and therefore, the key must be destroyed. The trusted configuration effectively trusts that the key has been destroyed. But how do you know if it's destroyed? Unlike a physical object you can see, destroyed digital objects can always keep a copy or store it somewhere. Therefore, a basically reliable

configuration means you need to trust someone or a group of people to destroy the key. If they didn't destroy it or if this ceremony was somehow hidden, someone has that key and can create money out of nothing. **Sigma** does not require this type of configuration because anyone who wants to help to destroy part of the ring can participate. **Zerocoin**, implemented by **Zcoin(Firo)**, uses a reliable configuration performed by third parties in an academic challenge called RSA Factoring Challenge in 1991, where the incentive to insert a backdoor is low and there was a considerable reward for breaking it. Although this is a decent implementation and with little chance of being compromised, we believe that the whole purpose of the blockchain is to build systems that do not require trust, and that same principle also applies to our privacy system.

ENHANCED SECURITY

Sigma's safety evidence is fully documented with much simpler construction, making it easier to audit. Sigma removes the reliable configuration and reduces the test sizes from 25 kB to 1.5 kB. The construction of Sigma does not suffer from the same flaw as the Zerocoin Protocol. The Sigma protocol allows users to prove that they have complete privacy in transactions with no reliable configurations through zero-knowledge cryptocurrencies.

ZERO-KNOWLEDGE PROOF (ZKP)

The concept behind the zero-knowledge test is a unique method where a user can prove to another user he knows an absolute value, without transmitting additional information. Here, the tester can prove that he knows the X value for the verifier without giving him any information other than the fact that he knows the X value. The main essence behind this concept is to prove the possession of knowledge without revealing it. The main challenge here is to show you know a "y" value without saying what "y" is, or any other information. If a user wants to prove a statement, he must know the secret information. In this way, the **verifier** could not transmit the information to others without actually knowing the secret information. Thus, the statement must always include that the taster knows the knowledge, but not the information itself. With that, you cannot say the value of "y", but you can say that you know "y". Here, "y" could mean anything. This is the central strategy of applying for the Zero-Knowledge Test. Otherwise, they will not be **Zero-Knowledge Proof applications**. That is why experts consider the applications of the Zero-Knowledge Test as a special case in which there is no

chance to transmit any secret information. The Zero-Knowledge test must have three different properties to be fully performed.

1- Completeness. If the statement is really true and both users follow the rules correctly, the verifier will qualify the transaction with no outside help.

2- Solidity. If the statement is false, the verifier will not allow the transaction to take place in any scenario. (probability of falsehood is equal to zero).

3- Zero Knowledge. The verifier does not store any information

Elysium

The **Elysium** protocol is a fork of the **Omni protocol** https://www.omnilayer.org. Elysium provides a **fully-decentralized** assets **platform** based on blockchain technology for creating and trading custom digital assets and currencies, facilitating the use of smart contracts, personalized **currencies/tokens** and even decentralized exchange functions. This layer expands the utility and functionality of the **5G-CASH** blockchain so as not to affect its core functions as a hybrid digital currency.

TOR

TOR (initially The Onion Router) is an open-source software developed several years ago by the United States government, for the military, and later released for use by the population, **TOR** briefly creates **encrypted tunnels** of traffic overlying the internet, to provide privacy to the user. The Tor community with the Crypto community, shared the ideal of privacy and decentralization. And in 2017, researchers from the University of Waterloo and the University of Concordia, both from Canada, introduced a system based on blockchain technology using onion routing techniques to facilitate anonymous deliveries. Using the **TOR network** protects your IP address and the origin of the transaction with a deep level of anonymity, and as the number of blocks increases, more nodes are added, which makes the network increasingly flexible and secure. In a simplified way, the system works as follows: Within the network, the TOR protocol finds an Entry Node in the network (or Entry Node) which is the initial connection node to the encryption protocol. The **Entry Node** is the place where transaction data will enter the TOR network securely and anonymously. Between your computer and the Entry Node,

a TLS **(Transport Layer Security)** tunnel is created. This tunnel is highly secure, no one can see what is going through it, all network traffic is encrypted from end to end. It connects to another node within the network (**Secondary Node**) where a secure connection is established between two nodes creating a new cryptographic key (Key 2). This Secondary Node connects to another node (Output Node) that will be where the data will leave the TOR network. The Secondary Node will then generate a new **cryptographic key** (Key 3) between it and the Outgoing Node, making sure that every transaction between them remains encrypted and secure. Upon completion of all connections and data transactions between nodes, the Outgoing Node sends a request to its destination address, stating that all data has been individually encrypted by each Node. The server that received the request will know only that the request came from the **Outgoing Node**, but it will not be possible to track the route of connections and information exchanges traveled between other nodes in the network. Consequently you will not know where the initial transaction was sent from. The final result obtained within the chain is that each node will know only the request sent through the node before its connection and the Login Node (which is the initial connection node) recognizes only your computer but does not know the destination of the data. This way, the network encodes its IP addresses between different connections, making tracing or identifying the principle of the transaction invisible.

Dandelion++

Lightweight Cryptocurrency Networking with Formal Anonymity Guarantees

Dandelion++ is a lightweight and straightforward network layer solution with formally guaranteed anonymity that can easily be implemented with existing cryptocurrencies. It explicitly improves upon idealistic assumptions of the original Dandelion proposal and differs from most broadcast communication anonymity protocols in its approach of usage goals and analysis metrics.

To better understand how **Dandelion++** works, it is essential to focus on how transactions are broadcasted in Bitcoin and how the original Dandelion protocol

worked. In **Bitcoin**, when a user broadcasts a transaction from a node, it is propagated to the nodes connected to that specific node known as its peers. The transaction message is then **subsequently propagated** in a chain reaction where each node further spreads the message to nodes that they are connected to. This is referred to as Bitcoin's gossip protocol and is how transactions can reach the majority of nodes in the network very quickly.

5G-CASH now implements a form of broadcast known as **diffusion** where each node spreads transactions with exponential and independent delays to its neighbors to mitigate against the deanonymization of a user's IP address. While effective, diffusion has recently been proven in several studies not to provide adequate anonymity protection.

The origin of a transaction message and its IP address (which is not included in a Bitcoin transaction message) can be mapped by third-party observers if they control enough nodes or use a supernode that is connected to a significant number of nodes. They can effectively map the originating address by observing which nodes see the transaction first. The Dandelion++ paper explicitly identifies how a study that used a supernode logged the relayed traffic of all the P2P nodes and observed the patterns of the transaction spreads over time to eventually deduce the source IP address. By linking the IP address with the pseudonym of the sender, a third-party can deanonymize users and link further transactions even if a new public key is used for each transaction.

Dandelion was initially proposed to mitigate these vulnerabilities but relied on theoretical guarantees that did not hold up in practice. The original Dandelion proposal made 3 idealized assumptions:

- All nodes obey the protocol
- Each node generates precisely one transaction
- All Bitcoin nodes run Dandelion

These assumptions clearly did not work in practice and are why Dandelion++ sought to address them. The original Dandelion protocol works in 2 phases:

1. Stem Phase
2. Fluff Phase

The **stem phase** is the anonymous phase where the protocol is designed to reduce the possibility of mapping back to the original node's IP address. In the stem phase, rather than a node broadcasting a transaction to all of its connected peers, it relays the transaction message through a privacy graph to a single random peer based on an algorithm. Subsequently, that node then only transmits the transaction message to another single peer, and the pattern continues until eventually (and randomly) one of the nodes broadcasts the message in the typical format of diffusion to the rest of the network.

This is where the **fluff phase** begins. Once a single node broadcasts the message using the diffusion method, the transaction message is propagated to a majority of nodes in the network quickly. However, it becomes much more difficult to trace back to the original node since the transaction message was transferred to many individual nodes through a privacy graph before being propagated in a manner that would allow an observer to map it to a single node. Instead, an observer could only map the spread of transactions back to the several nodes where the message was transferred in the stem phase, thus muddling the actual identity of the sender. In effect, this is abstractly similar to how a ring signature obfuscates the actual signer of a transaction.

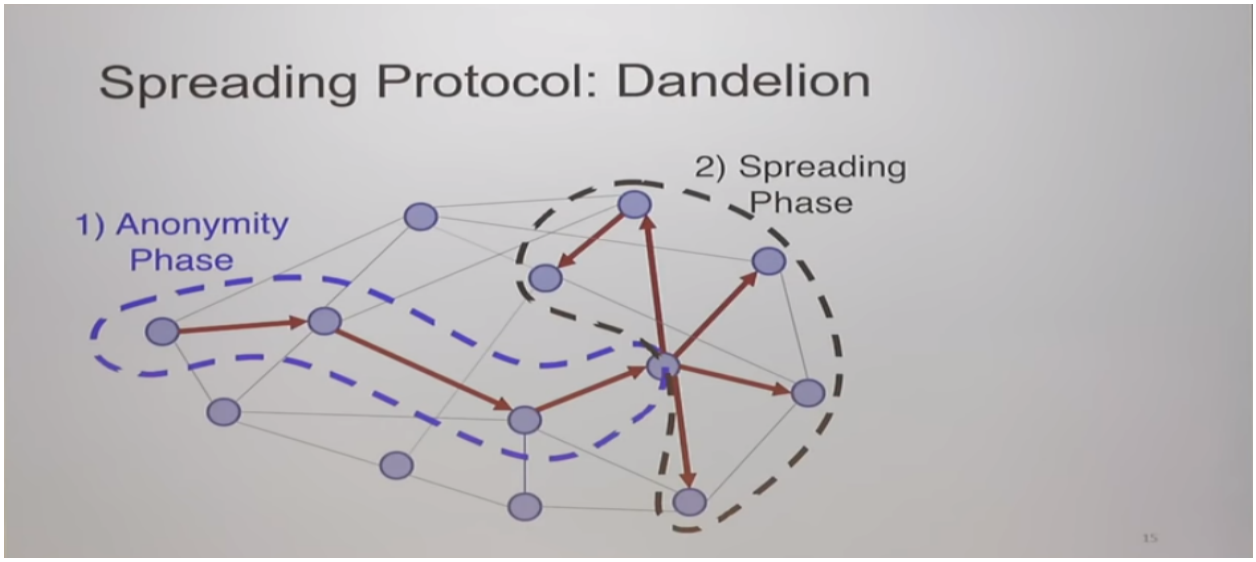

Image Credit – Giulia Fanti's Presentation in Lisbon

The Zcoin blog provides an excellent example of how the Dandelion protocol works by using typical high school gossip:

STEM PHASE

- Kathy: "Pssst, I have a massive crush on Nuwa. Please don't tell anyone"
- George: "OMG, did you know what Kathy told me? She has a massive crush on Nuwa. I only told you, please don't tell anyone"
- Alice: "Betty, you won't believe what Kathy's best friend, George just told me, Kathy is crushing hard on Nuwa! You're my best friend so I only told you, please don't tell anyone okay!"

BEGIN FLUFF PHASE

- Blabbermouth Betty: "Oh wow hot news…I have it from good sources that Kathy has a huge crush on Nuwa…Please tell everyone this is so exciting!"

The primary issues with the original Dandelion protocol stem from its underestimation of specific types of adversaries due to assumptions of their limited knowledge. Dandelion++ particularly focuses on making subtle changes to

the implementation choices of Dandelion such as the graph topology and mechanisms for forwarding messages.

As a result, these small changes to the algorithm exponentially augment the problem state space for anonymity analysis. Dandelion++ relies on increasing the amount of information that adversaries must learn to deanonymize users.

Dandelion++ notably **differs from Dandelion in its stem phase** where it passes transactions over intertwined paths known as cables before diffusing the transaction message to the network. The cables can be fragmented, but its intuition in selecting a node to propagate to is still confined to its local neighborhood. This is an important consideration when comparing network-level anonymity solutions like Tor that is an onion routing protocol where clients need global, current network information to determine transaction paths.

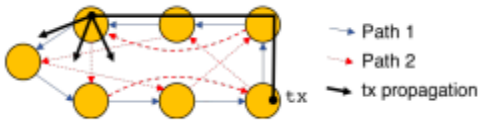

Fig. 2. DANDELION++ forwards messages over one of two intertwined paths on a 4-regular graph, then broadcasts using diffusion. Here, tx propagates over the blue solid path.

Image Credit – Dandelion++ Academic Paper

Both **Dandelion and Dandelion++ proceed in asynchronous cycles**. Each node advances when its internal clock reaches a certain threshold. For each period, Dandelion++ functions in 4 primary components with slight optimizations:

1. Anonymity Graph
2. Transaction Forwarding (own)
3. Transaction Forwarding (relay)
4. Fail-Safe Mechanism

Anonymity Graph uses a random 4-regular graph rather than a linear graph for the anonymity phase. The choice of Dandelion++ relays by nodes is independent of whether or not their outbound neighbors support Dandelion++.

Transaction Forwarding (own) is when every time a node generates a transaction of its own, it forwards the transaction along the same outbound edge in the 4-regular graph. This differs from one of the problematic assumptions in Dandelion where nodes are assumed only to generate one transaction.

Transaction Forwarding (relay) is the moment of probability in the stem phase where a node receives a stem transaction and either chooses to relay the transaction or diffuse it to the network. The choice to diffuse transactions to the network is pseudorandom. Further, a node is either a diffuser or a relay node for all relayed transactions.

Fail-Safe Mechanism is where for each stem phase transaction, each node tracks whether it is seen again as a fluff phase transaction. If not, the node diffuses the transaction.

The slight tweaks to the algorithm in these stages make it drastically more difficult to map IP addresses from observing the spread of transaction messages. The Dandelion++ **paper** goes on to identify specific attacks that could be used against the original Dandelion protocol including graph-learning attacks, intersection attacks, graph-construction attacks, and black hole attacks. With each attack vector, they demonstrate how Dandelion++ mitigates them with theoretical analysis and simulations.

Dandelion++ does not significantly increase network latency, and its practical feasibility was demonstrated on Bitcoin's mainnet. It provides a lightweight and effective network layer anonymity tool for reducing the possibility of mapping attacks to deanonymize users. Despite its advantages, Dandelion++ does not explicitly protect against ISP or AS-level adversaries which can use routing attacks to deanonymize users.

Dandelion++ vs Tor

Dandelion++ has some notable advantages over other network anonymity implementations such as Tor. Tor is the most prominent network overlay layer focused on privacy and it uses onion routing to conceal users' geographic location and IP addresses.

Tor's integration at the network stack level of cryptocurrency systems is exceedingly challenging. Monero is an excellent example of this as it has taken over four years to implement its Tor-like I2P Kovri project into their network and it is still a work in progress. Many cryptocurrency networks do not have the time nor the technical expertise to integrate this type of functionality.

Users routing their transactions through Tor is also not particularly feasible for mainstream Bitcoin users who are either unaware of the privacy deficiencies of the network or lack the experience to route transactions through Tor properly. Furthermore, Tor can be slow due to limited bandwidth compared to Dandelion++.

The same study that identified some deanonymizing concerns of diffusion broadcasting in Bitcoin also highlights attacks on nodes where they end up rejecting or blacklisting Tor connections. This can lead to deanonymizing transactions and mapping user IP addresses too.

Dandelion++ Simulations

https://github.com/gfanti/dandelion-simulations

POW Migration from X16R to X16Rv2

https://www.mdpi.com/2227-7390/8/8/1343/pdf

The X16R algorithm was upgraded into **X16RV2** by **Raven Coin** on the 1stof October 2019 to make it increasingly ASIC resistant while increasing the hash rate.

The Tiger algorithm was introduced into three parts of the X16R algorithm to finally get the X16RV2 version. The Tiger hash is designed to perform before the three algorithms **Luffa512, Kecca512, and SHA512**. The previously used X16R used 16 different algorithms operating in chain fashion, and the ordering was dependent on the last 8 bytes of the hash of the previous block. The reason why the move from X16R to X16RV2 was made was that an **ASIC 4x16 mining machine** was about to go online. There are also secret **FPGA** crypto mining farms on X16R. Numerous issues were solved by the introduction of **X16RV2** supporting all video cards. The main disadvantages are the huge expenses, **"uselessness"** of calculations and attack of 51%. A 51% attack, or majority attack, is a case when a user or a group of users controls most of the mining power. Hackers are given enough power to control most events on the network. They can monopolize the generation of new blocks and receive rewards, as they can prevent other miners from completing the blocks. The 51% attack is not a profitable option. It requires an enormous amount of mining power. And, since all the activity is exposed publicly, everyone can become aware of everything, the network is considered compromised, which leads to the departure of users. This will inevitably bring down the price of the cryptocurrency. As a consequence, the funds lose their value.

PROOF-OF-STAKE V3.0

POSv3 was invented to solve many of the problems created by **Proof of Work**. Proof of Stake's security has proven itself to be reliable & effective over years of testing. The latest robust **POSv3** has solved the issues faced with Coin-Age, Block Reward, Blockchain Precomputation while keeping nodes connected to the network and discouraging inactive nodes.

MASTERNODES

What Is a Master Node?

Master nodes are part of the infrastructure that sustains cryptocurrencies such as Bitcoin, Ethereum, and Dash. Unlike regular nodes, master nodes do not add new

blocks of transactions to the blockchain. Instead, they verify new blocks and perform special roles in governing the blockchain.

KEY TAKEAWAYS

- Master nodes verify new blocks of transactions in a cryptocurrency but unlike other nodes do not submit new blocks to the network for verification.
- Master nodes operate on a collateral-based system, meaning the operators need to own a significant amount of the cryptocurrency.
- In exchange for their investment in time and money, master node operators are rewarded with guaranteed crypto earnings, usually a percentage of their stake.

Understanding Master Nodes

There are several types of nodes that together form the infrastructure of a decentralized blockchain, collectively providing transparency and security and running the software that implements a cryptocurrency's rules and functionality. Nodes maintain the massive ledger of public transactions in a given cryptocurrency and verify new transactions. Master nodes also play a special role in the management and governance of the blockchain's protocol.

Operating a master node requires a significant financial investment and running costs, including a significant stake in the cryptocurrency itself and computer hardware that is far more expensive than your average laptop. It also requires expertise. As an incentive for people to maintain master nodes, operators are rewarded with cryptocurrency earnings, usually a share of block rewards.

DARK GRAVITY WAVE WHAT IS DGW?

People often ask what Dark Gravity wave is, and how this difficulty retarget algorithm works. **Dark Gravity Wave** is an open source mining difficulty re-adjustment algorithm developed by Evan Duffield (creator of X11/Darkcoin/Dash). **DGW** came after KGW (Kimoto Gravity Wall), the most popular difficulty retarget algorithm that adjusts difficulty every block using information from the previous blocks. Dark Gravity Wave was inspired and is

based on Kimoto Gravity Well (**KGW**). Also DGW is proven to reduce some theoretical disadvantages of KGW such as time-warp exploit. Also there are 2 other difficulty adjustment algorithm namely Nite's Gravity Wave and Digishield. You will come across all these algorithms only in Proof-of-Work mineable coins and not on **POS** coins. In Bitcoin the standard block difficulty readjustment is set to adjust only every 2016 blocks. The problem with this scheme is that it gave rise to multipool mining. Multipool Mining is a process of jumping from one crypto to another mining the most profitable one at that current moment. Then the miners dump the mined coins to buy back Bitcoins. True, this actually happened back then when the price of Bitcoin Cash (BCH) arose. Miners will only focus on economic incentives; as BCH became more profitable miners almost abandoned the Bitcoin network to mine BCH. Once BCH adjust its difficulty miners will then jump back to mine Bitcoin. People actually thought it was 51% attack but it's actually nothing but a seesaw of hashing power being delivered between Bitcoin and Bitcoin Cash based on their profitability. This was a serious problem with Bitcoin and this is what gave birth to Dark Gravity Wave and other mining difficulty regulators. **BENEFITS OF DARK GRAVITY WAVE DGW** uses multiple exponential moving averages and a simple moving average to achieve the smoother difficulty re-target mechanism. Coins that have Dark Gravity Wave as their difficult algorithm are immune to issues like multipools as it retargets difficulty every single block. Not just that; with DGW the chain becomes more secure and block times are much more consistent; despite large fluctuations in mining power. Apart from controlled difficulty some other benefits of Dark Gravity Wave are security, faster transaction, more miners and reliable chain.

Vidulum V-Staking (VRS)

V-staking is a partnership in which the 5G-CASH and Vidulum team got into a contract to allow people accumulate VGC coins in the non custodial Vidulum web or mobile wallet using the Vidulum Reward System (VRS) and earn VDL coins based on their VGC balance percentage.
https://vidulum.app/blog/2021/03/20/5GCash-VGC-Is-Live-On-Vidulum-App/

WHY WOULD YOU NEED A TIP BOT?

Send cryptocurrency tips to anyone on Discord & Telegram

tip.cc aims to make sending and receiving crypto easy even for cryptocurrency newbies. Send coins to users who don't even have a wallet!

- no minimums and no fees on tips
- tip multiple users at the same time with no transaction fees
- make airdrops, rains and other community-engaging activities

As VGC is included in the **tip.cc wallet service** it can be **tipped** and **airdropped** all over **Discord** making it easy to be spent for any good or service in which **5G-CASH core Team don't participate by any means**. All transactions spent for purchasing a good or a service is tied to the parties involved.

5G-CASH Discord Staking Bot

https://github.com/5G-Cash/cryptocurrency-crypto-bot

5G-CASH Staking Bot is a Node.js open-source wallet bot for Discord in which everyone is welcome to use its features for **FREE in our Server**. The purpose of having such a service is to allow everyone to earn VGC rewards from the Staking Pool with a minimum deposit of 1 VGC coin so everyone can participate when they are not able to stake on their own.

How secure are users' balances?

Bot’s Database is backed up every hour to keep people’s balances safe and secured.

Links

Website: https://fiveg.cash
Explorers: https://explorer.fiveg.cash
https://blockbook.fiveg.cash
Github: https://github.com/5G-Cash/5G
Pools stats: https://miningpoolstats.stream/5gcash
Price Tracking: https://www.coingecko.com/en/coins/5g-cash
https://coinpaprika.com/coin/vgc-5g-cash/
Facebook: https://www.facebook.com/5gcashblockchain
Twitter: https://twitter.com/5gCash
Chat: https://discord.com/invite/uXHvyBT
Telegram: https://t.me/VGC_5GCASH
Bitcointalk ANN: https://bitcointalk.org/index.php?topic=5264647

Third Party Mobile Wallets
IOS: https://apps.apple.com/us/app/id1505859171
Android:
https://play.google.com/store/apps/details?id=com.vidulumwallet.app