HYCON WHITEPAPER v1.3

INFINITY PROJECT

ABSTRACT

This paper begins by outlining the vision of the Infinity Project, planned for a 3 phase roll-out: 1) the ‘HYCON’ coin 2) the open-source Infinity Platform for customizable enterprise blockchain solutions, and 3) a decentralized cryptocurrency exchange platform. However, this whitepaper’s main objective is to provide a detailed analysis of ‘HYCON’, a fast and secure cryptocurrency that makes use of the SPECTRE protocol to enable high transaction speeds while maintaining security. Identified herein are a sample of the challenges and limitations of many existing cryptocurrencies and solutions proposed by the HYCON coin. Additionally, the technical specifications of HYCON will be introduced, as well as a brief discussion of SPECTRE and its implementation in this project.

INTRODUCTION

> "...cash, after millennia as one of mankind's most versatile and enduring technologies, looks set over the next 15 years or so finally to melt away into an electronic stream of ones and zeros."
>
> **The Economist (2007)**

In today's world of electronic and mobile banking, money is transforming from tangible notes and coins into intangible digital numbers that are moved around the Internet. It is a natural progression, in this environment, for new forms of currency to emerge, which exist only as cryptographically secured strings of digital code, known as 'cryptocurrency'. This digital currency revolution began in 2008 when the still anonymous Satoshi Nakamoto published the Bitcoin whitepaper [19].

Today, new cryptocurrencies are released on an almost daily basis, sharing one unifying concept: the underlying technological architecture that is the blockchain.

A blockchain itself is a shared public ledger of all transactions made on the system - from the genesis of the first block until its conclusion. The ledger - the blockchain defined above - is built using a linked list, or chain of blocks, where each block contains a certain number of transactions that are validated by the network within a specific timespan.

The Infinity Project will introduce a new cryptocurrency called HYCON, designed to address some of the challenges faced by existing blockchain technologies. What follows is an outline of the existing state of blockchain development with a focus on the problems that need to be addressed, the goals of the Infinity Project itself, how HYCON proposes overcoming existing blockchain limitations, and the technical specifications of HYCON.

DISCUSSION OF EXISTING BLOCKCHAIN TECHNOLOGIES

For the purposes of this discussion, the Bitcoin and Ethereum blockchains will be the focus as they are the most widely used and well-studied implementations of blockchain technology to date.

A useful reference point for examining blockchain technology is the work of Yli-Huomo et al.[31]. Presented therein is a comprehensive summary of recent work on blockchain technology and the limitations inherent in a blockchain based system. While their research was focused entirely on papers discussing the bitcoin blockchain, the findings presented are broadly applicable for the purposes of this discussion. The key discussion metrics used were drawn from Swan [29] and again, these are applicable here.

This research highlights seven limitations of current blockchain systems:

- Throughput
- Latency
- Size and Bandwidth
- Security
- Wasted Resources
- Usability
- Versioning, Hard Forks, and Multiple Chains

Throughput

A representative blockchain such as Bitcoin takes 10 minutes or more to confirm transactions resulting in a maximum throughput of 7 transactions per second with current transaction rates in the region of 4 per second. Ethereum can process 10 or more transactions per second, with confirmation times approximately 10 times faster than those found on the bitcoin network. However, benchmarking the current throughput capacity of these blockchains against the VISA network is a useful comparison to understand the current limitations of these blockchains. The VISA network can verify transactions in seconds, processing 2000 transactions per second on average, and features a maximum capacity of 65,000 transactions per second.[30] As can be seen by these metrics, there is still a large discrepancy in the throughput of today's most used blockchain networks versus traditional, centralised payment networks like VISA.

A primary factor limiting the throughput of blockchain networks is the latency between nodes. While there have been some promising attempts to address this problem, such as the lightning network [22] due to be adopted by Bitcoin, and the Raiden network [23] that is already running a micro version on the Ethereum blockchain, consensus has yet to be reached on a viable, long-term solution.

Latency

As mentioned above, latency and throughput go hand in hand as limiting factors for the fact that maximum throughput of a network is limited by the latency between nodes. If there is a high latency between nodes, the risk of mining on an old block is increased. On the Bitcoin network, for example, the average time taken for a block to propagate to 50% of the nodes is just under 2 seconds, with 90% of nodes reached after approximately 13 seconds (as of April 2017) [4]. For Ethereum, the average time for propagation to 50% of nodes is less than 1 second, with 90% of nodes reached in approximately 10 seconds [11].

For Bitcoin, the ratio of the block mining time to the network propagation time is large, meaning that latency between nodes does not act as a large limiting factor. With Ethereum's shorter inter-block time, the wasted time could be more problematic, but Ethereum uses an algorithm based on the GHOST protocol [27] to incentivise mining on the longest chain instead of attempting to continue to mine on parallel chains that were created due to high latency, or low inter-block times.

Size and Bandwidth

There are two considerations that must be addressed in a discussion regarding size and bandwidth: the physical data that represents the full blockchain, and the size of individual blocks being sent over the network. With the requirement that a *full-node,* which can mine new blocks and interact with the blockchain network, must maintain a local copy of the entire blockchain, it is clear that the storage capacity required to maintain this local ledger is directly proportional to the number of blocks on the chain, thus leading to a much higher risk of centralization when the blockchain becomes so large that only a few nodes are able to process a block [15]. In addition, when transaction volume starts to push the limits of the available bandwidth, coupled with a hard-cap imposed by the block size, mining fees can increase significantly, which could lead to changes in the core protocols to allow for greater transaction volume: i.e. larger block sizes, or shorter block times. In this eventuality, a hard fork, which is generally considered undesirable, would be required to make the necessary protocol upgrades.

Security

One of the biggest selling points of POW blockchains as a technology is the fact that they are very difficult to hack. In order for a fraudulent user to alter an entry that is already present on the blockchain, they would need to redo the proof of work for that block as well as all subsequent blocks that makes reference to it. The computational resources needed to successfully perform an attack of this nature is equal to the hashing power of 51% of the network, hence the name, "51% attack". However, this is an unlikely prospect as the mining benefits of possessing 51% of the network would outweigh those to be gained by acting fraudulently.

A second method of attack is referred to as a Sybil Attack, where a fraudulent entity creates multiple fraudulent identities for use on the network, which then try to subvert the network to their advantage. In a POW based system, such as Bitcoin, your ability to influence the network is determined by how much hash power you use to find new blocks. Pretending to be two miners would require the hash power to be divided, resulting in no advantage.

There are, however, other ways to attack user accounts on a blockchain network. Often, users rely on the safe storage of their private keys by centralized exchanges, which, if compromised, can provide an attacker with access to their wallets, and by extension, their cryptocurrency.

Another security risk in the blockchain space is found in the implementation of *smart-contracts* that feature coding errors. A particularly well-known and successful exploitation of a smart contract occurred on June 17, 2016 and is now colloquially known as the "DAO attack", wherein an attacker was able to use a small flaw in the code that executed the smart contract to obtain an estimated $50~60 million worth of ether, an event that eventually led to the controversial hard fork that split the Ethereum network in two, creating Ethereum Classic. [2]

Wasted Resources

The electrical needs and by extension environmental impact of the bitcoin blockchain is considerably large. At current estimates, 249 kWh of electricity is required to validate a single transaction, with upwards of 32 TWh used by miners annually to continually add new blocks to the Bitcoin blockchain.[6] While the figures involved are lower for Ethereum, the energy outlay, and thus environmental impact, is still massive.[7] In fact, if you were to combine the amount of energy used to protect the Bitcoin and Ethereum networks, there would be enough to power the country of New Zealand for a year. There are currently movements away from proof of work blockchains, with Ethereum being the most prominent proponent for a move towards proof of stake.

Usability

On the Bitcoin blockchain, transactions are published as part of a block approximately every 10 minutes, and it is standard to wait up to 50 minutes or more after each transaction has been published to allow for subsequent verification. Taking a real-world example, this would be analogous to attempting to pick up groceries from a store and then having to wait in line for an hour while your payment is processed. This is clearly unacceptable for real-time application to a real-world use case.

A second, more nuanced concern with Bitcoin, and the majority of currently available cryptocurrencies, is the concept of anonymity, or pseudonymity as is the common case. Transactions are public and shared by all participants on the blockchain. This is not always desirable for transactions where discretion is required as algorithms are able to extract transaction data from a user's "private" transactions. To again use a real world example for illustrative purposes, suppose a user sends money to their mother; based on the transaction information, it is possible to see: 1) How much bitcoin has been sent and received by each user now and throughout the entire history of their addresses, 2) The balances of both addresses at any given time in history, and 3) To which other addresses each user has sent funds to or received funds from. Essentially, senders and recipients of transactions can see the financial history of the other, and can even have the ability to know what was bought, what was gambled on, or even which entity received 'anonymous' support, once you are able to identify the address of an individual. As the American FBI has proven several times now, Bitcoin is not truly anonymous. For many users, financial transparency is perhaps one of the largest disadvantages of using Bitcoin. However, researchers are working on fixing this problem with solutions such as zkSNARKS [24] (zero-knowledge cryptography), which is a

privacy mechanism built into ZCash, and added to Ethereum with the Metropolis (Byzantine) upgrade.

Versioning, Hard Forks, and Multiple Chains

The primary problem of forking on a blockchain is the loss of consensus and security. Take the hyperbolical example of a severely bloated blockchain which uses 100% of the computing power available in the universe, and a contrasting example wherein 100 competing chains each possess 1% of the available computing power.

In the case of the first example, a 51% attack would truly require 51% of the computing power available in order to overtake the chain maintained by honest nodes, however in the fragmented case, only 0.51% of the computing power available in the universe would be required to corrupt any individual chain.

Blockchains rely on consensus being maintained by having the combined computational power of honest nodes outweigh the power available to malicious ones. With a fragmentation of the chain and reduced computational power being applied to each fork, an attack has more chance of being successful because less hash power is required to carry out an attack.
Hard forks are another generally undesirable outcome that often result from a loss of protocol consensus. Ideological differences between different stakeholders within a given blockchain ecosystem can lead to a split, or a fork of the chain, with well-known examples being the breakaway of Bitcoin Cash due to Bitcoin's scaling issues and inability to be used as a fast and cheap means of electronic cash. The previously referenced Ethereum Classic also forked away from Ethereum on the philosophical basis of blockchain immutability. However, hard forks are not always contentious and often come about through changes in core protocols of a blockchain system, such as Ethereum's 2017 Metropolis upgrade. After a hard fork, the overall hash power that was applied to the original chain may still be there. However in the case of an ideological hard fork, the split between two competing chains, could possibly leave them less secure and more vulnerable to attack.

INFINITY PROJECT - CORE GOALS

During the formation of the Infinity Project we asked the following 2 key questions:

✓ Given current limitations of existing cryptocurrencies, what are the needs and wants of the market and how can we provide solutions?

✓ What properties are necessary for a cryptocurrency to be widely adopted and integrated into the wider economy?

With these questions in mind, we conducted a thorough analysis of existing blockchains - including Bitcoin, Ethereum, and a variety of promising Altcoins - to uncover the strengths and weaknesses of each project. However, it was difficult to find a project that satisfactorily addressed the issues posed above.

Therefore, the Infinity Project team began researching new technologies and algorithms that were more suitable for mass, real-world adoption, to help us meet our goals. At the same time, we started to design the basic framework of the Infinity Project, and formulated the following 5 core goals:

INFINITY PROJECT CORE GOALS

1. Identify actual market needs for a cryptocurrency
2. Develop a cryptocurrency that is flexible
3. Create user-centric blockchain platform
4. Develop an ecosystem that promotes sustainable innovation
5. Investigate methods for a decentralized cryptocurrency exchange

Core Goal 1 - Identify Market Needs

Although many blockchain projects have gained mainstream attention and recognition recently, no cryptocurrency has penetrated digital commerce on a global scale. More precisely, a great divide still exists between most crypto projects and real world solutions. Cryptocurrency acceptance and adoption is currently limited to a very small group of online merchants and a handful of other services, making it unfeasible to completely rely on Bitcoin or any other currently available cryptocurrency as the de facto digital currency of choice.

To help overcome this issue and expedite use cases and adoption, it is useful to work jointly with experts in a given field or community, along with developers, in order to drive the development of a successful, market-friendly currency that best serves all users.

Therefore, to answer one of the two key questions posed by our Infinity Project team, "What is the user-centric currency that the market wants?", we must define the core blockchain technology required in order to find a mutually desirable solution from both a market and

development standpoint. This led the Infinity Project team to conclude that the first key success factor (KSF) when developing our new cryptocurrency is to design and implement it based on the premise of providing practical solutions that the market needs.

Core Goal 2 - Flexible Currency

The Infinity Project team decided to step away from traditional viewpoints of monolithic monetary development present in many existing blockchain projects and introduce the concept of a flexible implementation platform that can incorporate various monetary models.

This led to the creation of our "Hyper-connected Coin" (HYCON), which from its very beginnings, was designed to be fast, cheap, scalable, and safe, and thus ready for adoption and usage in a variety of real-world situations.

The underlying Infinity blockchain that it's built upon has been designed with an interchangeable, modular structure which will facilitate the easy adoption and alteration of the underlying technology to suit specific needs.

Core Goal 3 - User-Centric Platform

Arguably, one of the most important parts of the paradigm shift that Bitcoin ignited is the facilitation of a secure, decentralized exchange of value electronically - one that is open to all, and opened the door to the once unrealistic notion of making payments without banks.

However, a steep learning curve from the conceptual level down to the actual UI and UX of most cryptocurrencies is one of the key hurdles currently hampering greater adoption. The Infinity Project reduces these barriers to entry by providing a more straightforward and user-friendly platform, in addition to an intuitive wallet and exchange platform interface. Ultimately, our goal is to allow more people to harness the paradigm-shifting power of blockchain technology.

Core Goal 4 - Flexible Innovation

One of the most important aspects considered during the development of the Infinity Project was how to help more people, businesses, governments, NGOs, etc., harness the power of blockchain technology. Thus, our Infinity Project team is implementing the Infinity Platform, a flexible blockchain platform concept that evolved from the study of various other existing blockchains, platforms, and cryptocurrencies. While HYCON is a part of the Infinity Platform, it will not be its sole component.

The goal of our Infinity Platform research is to create a platform that is intuitive to use and one that can be implemented in various ways. Example use cases for the Infinity Platform include: implementing a secure cryptocurrency based on HYCON that is fast and cheap to use as a means to exchange value; the creation of decentralized corporate ledgers to enhance information security and facilitate more efficient data storage and transmission; and adding cryptographic security to stock exchanges. Potential use cases and innovations built with the Infinity Platform are vast, and can adapt to offer potential users the flexibility to create the blockchain solution they need.

Core Goal 5 - Secure Decentralized Exchange

An active area of research for the Infinity Project is giving users the ability to exchange different cryptocurrencies in a decentralized manner. Current exchanges rely on centralization to cheaply and quickly trade cryptocurrencies, however this centralization requires users to entrust their fiat and cryptocurrency holdings to the exchange.

Unfortunately, the source code that powers these exchanges is often not publicly available for review, despite the huge volumes of transactions that pass through these exchanges. Globally, there have been multiple incidents where exchanges have had their user's cryptocurrencies stolen by malicious actors. The centralization of user funds and information existing on exchanges today will continue making these companies targets.

As part of the Infinity Project's future research, we intend to integrate the concept of atomic swaps into Hycon to allow for our currency to become a true medium of exchange. Through HYCON, multiple other cryptocurrencies will be tradable, and transaction fees will be distributed to the miners who protect the network. Atomic swaps will allow HYCON to be held in escrow pending the proof of payment of another cryptocurrency, and thus facilitate trustless P2P trading of HYCON and other cryptocurrencies.

HYCON TECHNICAL SPECIFICATIONS

Characteristic	Specification
Hash Function	Cryptonight & Blake 2b
Consensus Protocol	SPECTRE
Chain Structure	Directed Acyclic Graph(DAG)
Block Speed	1000ms
Mining Method	PoW

Genesis Block

The HYCON Genesis Block was published on January 4th, 2018, at 3:15am KST (GMT+9). It is available to be viewed on GitHub as part of Team Hycon's repository[36]. For more information about the Genesis Block, please see Appendix A.

Hashing Algorithms

The first version of this document cited Blake2b as the only hash function being used as part of the HYCON system. However, due to recent developments in ASIC technology[38], it was decided to move away from the Blake-2b hash and to instead use the ASIC resistant hash Cryptonight, which is also employed by Monero. What makes Cryptonight interesting is that it uses pseudorandom memory read/write operations as part of its hashing operation. This leads to performance being loosely comparable whether on a CPU or GPU, while also rendering it incompatible with standard ASIC architecture. In the future, and in order to deter centralisation of mining resources, it is planned to follow the example set by Monero and periodically tweak the hashing algorithm to maintain ASIC resistance over the course of the mining period. [43]

Consensus - SPECTRE Protocol

In contrast to the Nakamoto protocol used for consensus on the Bitcoin blockchain, HYCON implements a protocol called SPECTRE to maintain consensus [26]. SPECTRE generalises a blockchain into the form of a directed acyclic graph(DAG) by employing a voting algorithm between pairs of blocks in order to specify their order in a pairwise manner, i.e. block x should be applied before block y or block y should be applied before block x. While a full description of the SPECTRE protocol is beyond the scope of this whitepaper, a basic outline of the voting rules is provided below.

Voting Rules

In order to discuss the voting rules in SPECTRE, it is useful to refer to a visual representation of the process. It should also be noted that no votes are communicated between nodes and

there is no need to explicitly participate in a vote. Rather votes come from blocks and the way in which they vote is implied from the structure of the DAG.

The criteria used in the voting process in SPECTRE are as follows;
The important terms to be noted are $past(x)$and $future(x)$, which designate the blocks that are reachable from x, and the blocks which reference x as an antecedent respectively. More specifically, block y is in $future(x)$ if x is in$past(y)$. i.e.

$$y \in future(x) \Leftrightarrow x \in past(y)$$

It should also be noted that the virtual block, designated $virtual(G)$, is a hypothetical block that has the entire DAG as its past.

For some block designated *z* voting on some other blocks designated *x* and *y*:

1. If z is in $future(x)$ but not in $future(y)$ then it will vote in favour of x.
2. If z is in $future(x)$ and $future(y)$ the vote will be determined recursively based on the prospective vote of a virtual block with a past equal to $past(z)$
3. If z is not in $future(x)$ or $future(y)$, then the vote will be decided by the majority vote from the set of blocks designated by $future(z)$
4. If z is a virtual block with $past(Virtual(G))$ i.e. its past = the entire DAG, it will vote in accordance with the majority vote of the DAG.
5. For the cases $z = x$ or $z = y$, the block will vote for itself to be correct, provided that y is not in $future(x)$, or vice versa.

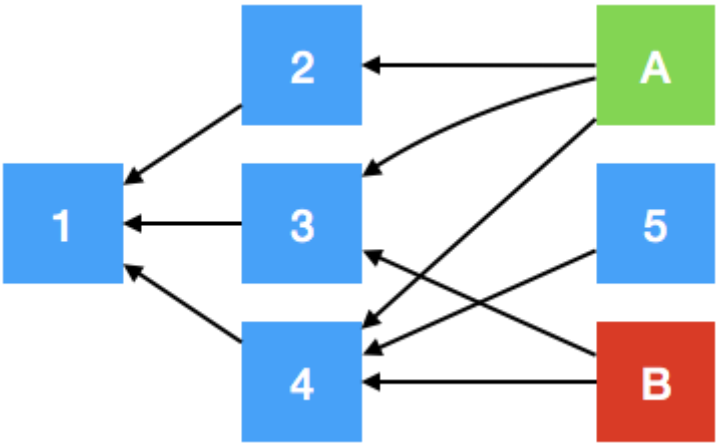
Application of SPECTRE Protocol to Example DAGs

To best illustrate the application of SPECTRE, it is useful to work through an example of the protocol in action step by step and provide snapshots of the state of the voting process as it proceeds. This particular example is drawn directly from the SPECTRE whitepaper. [25]

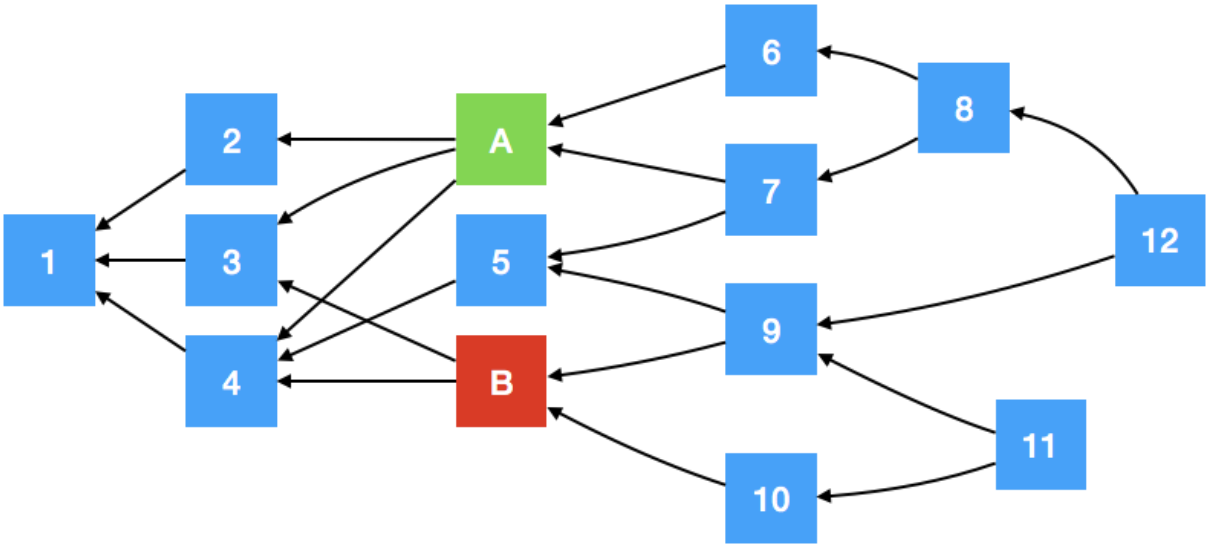
Example Case - A Double Spend

Take the simplest possible case where a block *A* contains a transaction *t1* and a second block *B* contains a conflicting transaction *t2*. These conflicts could be malicious, or merely caused by latency between nodes leading to transactions being published twice, such that two miners are collecting the same transaction fee. Depending on the structure of the DAG the result of this double spend will be resolved differently as the two blocks will have differing pasts and futures.

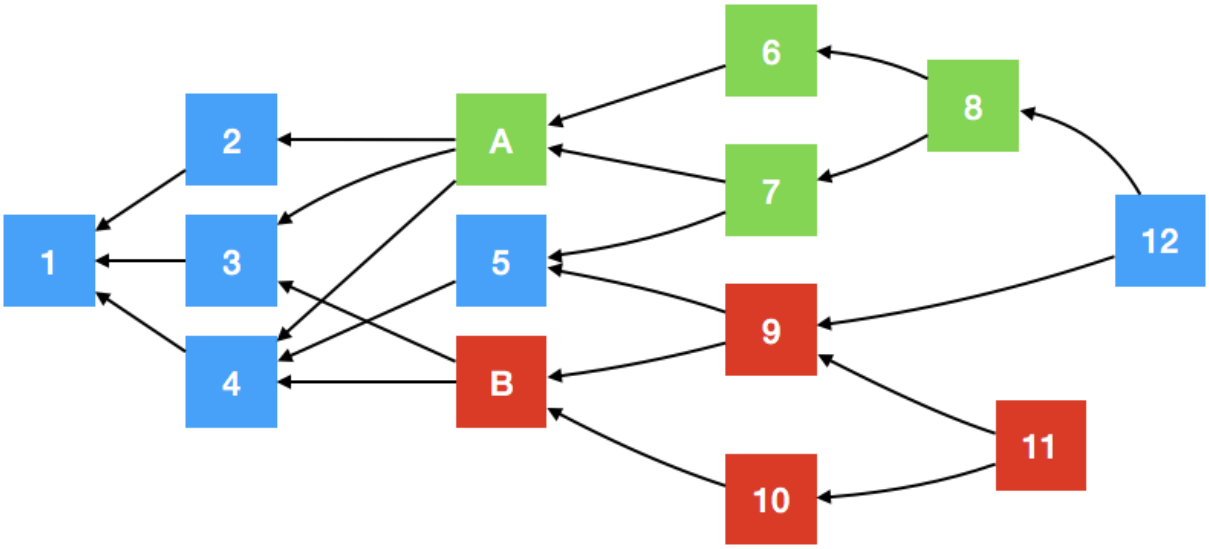
The initial case for this example looks as follows with blocks A and B being published at approximately the same time as block 5.



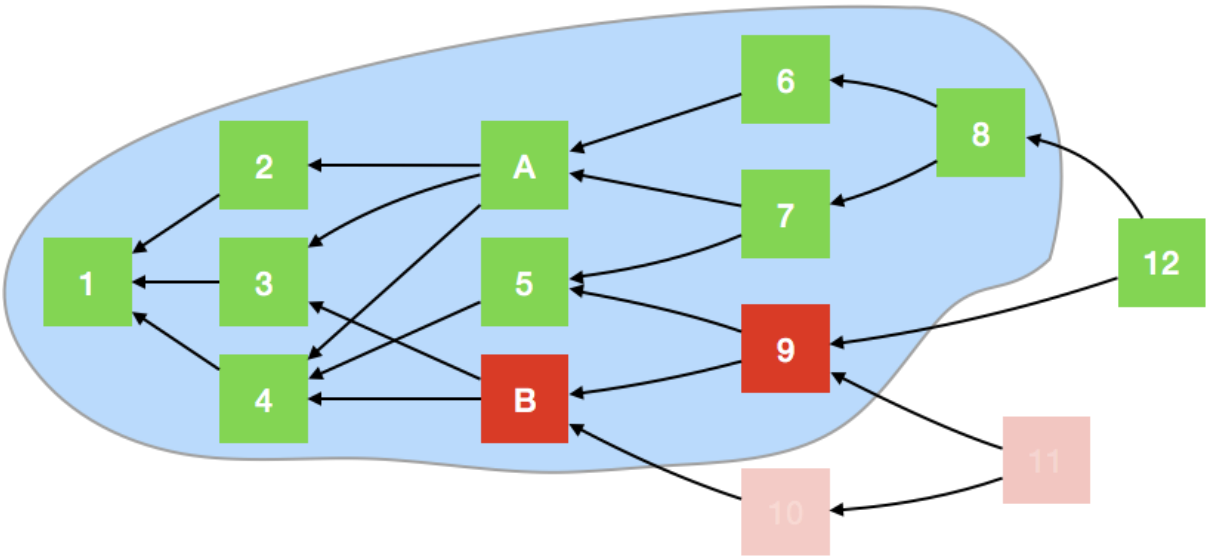
At this stage the system is unaware of the double spend, as no subsequent blocks have been published referencing both of the conflicting blocks. However, as the structure of the DAG develops and more blocks are added, the double spend is discovered and the structure of the DAG will be analysed to determine which block takes precedent.



In the above diagram, block 12 is the first block published with reference to the double spend between A and B. Following the rules listed above, the votes can be counted as follows. Blocks 6, 7, and 8 all vote for block A, because block B is not present in their past. Similarly, the votes of blocks 9, 10 and 11 are for B, for the same reason.



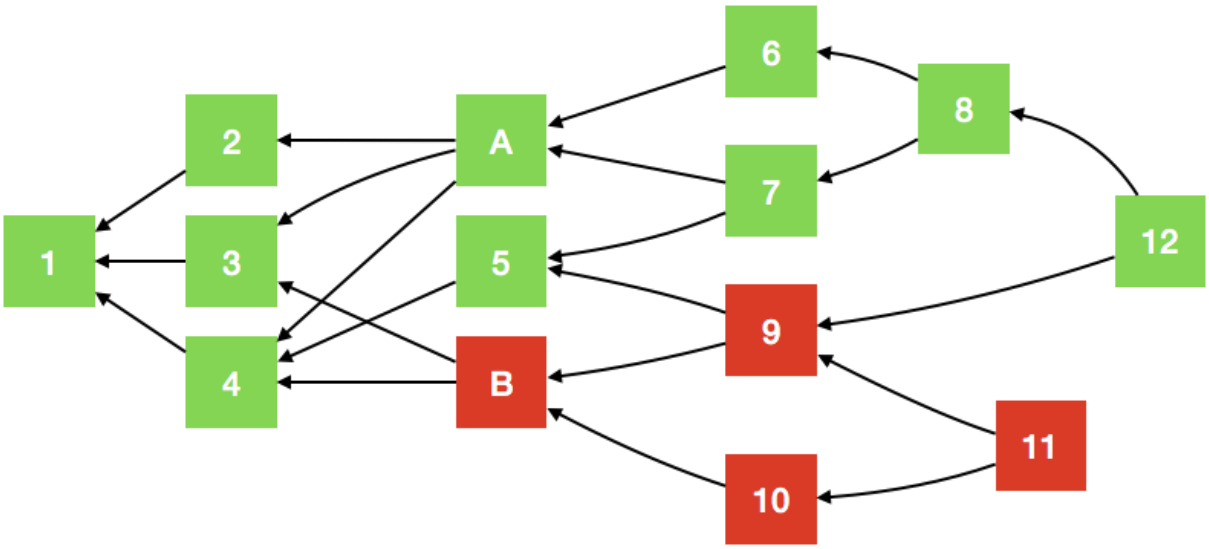
Block 12 votes based on a recursive call over its past. As blocks 10 and 11 are not included in $past(12)$, they are not considered when determining block 12's vote. The voting area for block 12 is shown below.



Blocks 1-5 are not in the sets of blocks $future(A)$ or $future(B)$, so they will vote the same as the majority of their futures. In the case of this recursive vote, these blocks all have more votes for block A in their future and so also vote for block A. Block 12's past contains 9 votes for block A and 2 votes for block B, So block 12 will vote for block A. If there had been a tie, Block 12 would break the tie deterministically so that all servers can agree on which way block 12 votes. Since we only use $past(12)$ to determine the vote of 12, its vote will never change.

The remainder of the voting process in the DAG is based on the future of the remaining blocks. Once Block 12's vote has been confirmed, Block 5 votes in favour of A because the votes of

7, 8, and 12 outweigh the votes of 9 and 11. Block 4 sees votes for A from A, 5, 6, 7, 8, and 12, with blocks B, 9, 10, and 11 voting for B, thus Block 4 also votes for A. It is a similar case for blocks 3, 2, and 1, which all cast their votes for A. Leading to a final vote tally for this voting procedure of 10 votes in favour of A and 4 votes in favour of B.



An interesting property of SPECTRE is that, especially in simple cases such as the one illustrated here, it replicates longest-chain selection models used in other blockchain technologies. Following the route from 1 -> 12 passing through A, and the same route but passing through B, it can be seen that the route 1->A->12 is longer than the route 1->B->12, i.e. the longest chain wins.

DAG Versus Blockchain

A common question that gets asked when discussing HYCON is about the DAG structure itself. The DAG structure employed in HYCON is a consequence of committing to a 1000ms block interval. While looking at ways that throughput on a blockchain could be increased it became apparent that system latency on a distributed network would be a limiting factor leading to unintentional forking of the blockchain. Rather than avoiding these forks, it was decided that it should be embraced in the resulting structure and the oft misunderstood and overemphasised DAG was borne.

The advantage of a DAG structure over a linear blockchain is primarily that it allows for shorter intervals between blocks, which in turn leads to higher transaction confirmation speeds. In contrast to currently existing blockchains where newly mined blocks are attached on to the end of the blockchain by referencing the hash of the previous block, DAG works by adding a new block with references the the DAGs current tips. This allows blocks to be published simultaneously from different nodes without running the risk of forking the chain. As new blocks are permitted to have multiple antecedents, they can be added concurrently and thus miners can still reap the mining rewards without having to worry about their blocks being orphaned. Where problems occur is if nodes publish transactions that have been simultaneously published elsewhere, leading to possible double spend. Using SPECTRE, it is possible to achieve consensus regarding which transactions should be rejected without orphaning the entire block.

INFINITY SPECTRE Implementation

The voting procedure in SPECTRE is quite a drain on resources, so its implementation needs to be carefully managed. Our initial prototype was written in Python for ease of development. However, the final version of the INFINITY SPECTRE implementation is to be written in Rust, such that complete control over data structures and memory management is maintained, allowing for better performance.

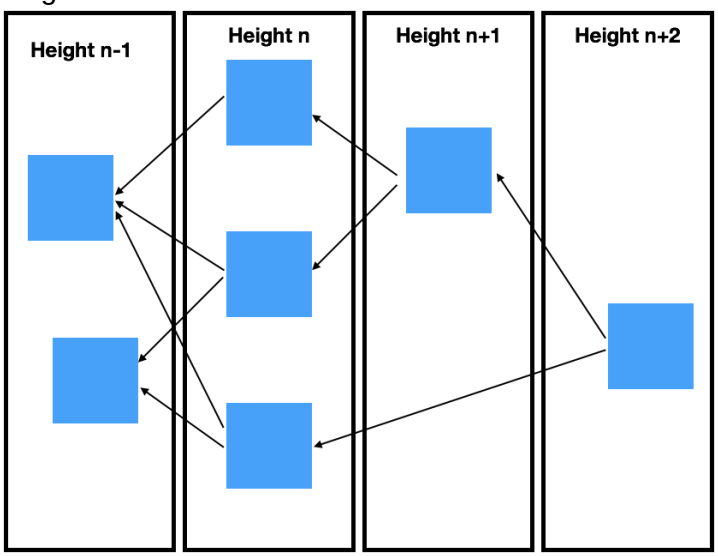
Block “Height” and Chaining

As can be gathered from a cursory glance at the DAG structure, the traditional idea of block height as used in Bitcoin or Ethereum requires a slight semantic modification. In those representative blockchains, the height represents the number of blocks that have been chained to the top of the genesis block. In HYCON the height is a more general descriptor representing the number of DAG layers existing above the genesis block. The calculation to be applied is quite simple. The height of the new block is one layer more than the height of its highest parent.

For any new block B with set of Parents P:

$$Height(B) = max(Height(p)) + 1;\ p \in P$$

Diagrammatically, this situation looks like the example provided below. Newly published blocks references the highest available unreferenced blocks, and have a height set as one layer more than the highest referenced block.



Network Infrastructure - Node.js, Typescript

The advantage to using this setup for the system architecture is Node.js’ inbuilt support for asynchronous operations. Node.js allows for cross-platform “non-blocking event I/O”, where individual components can wait for the results of operations outside the flow of normal operations. The waiting components are only triggered and executed on the occurrence of a certain event, such as a message being received from the network, or input from a user, allowing other codes to execute during the waiting period. [20]

The use of Typescript was decided upon due to the strong type checking that it enforces on what is essentially Javascript. Using a typed version of Javascript allows the development team to build a platform that takes advantage of the asynchronicity provided by Node.js while also making debugging a simpler process due to explicitly defined types. As the Typescript files require compiling before being run, many syntax and type errors can be caught easily at the compilation stage, rather than by trawling through a maze of callbacks step by step in the debugging process as is common in Javascript applications.

Serialization - Protocol Buffers

There are any number of messages flying around the network at any given time in a blockchain network. It is important that the node software is able to decode that data in a consistent and correct manner. The use of protocol buffers, developed by Google[14], allow for consistent message definitions that can be used across different platforms, to allow for nodes running the infinity blockchain to be developed in numerous programming languages. The serialization layer is language agnostic, which is highly useful for cross-platform applications. Protocol buffers also allow for backward and forward compatibility, increasing the chance that an update will only require a soft fork, rather than a hard fork. It also makes it easier for third party software to be compatible, allowing other developers to interact with the Hycon network.

Mining

Overview

Publishing of a block will require proof of work, similar to the majority of existing cryptocurrencies. Miners will calculate the hash of the next block based on the hashes of the tips of the DAG and include the merkle root of the transactions in the block, and a nonce, that will be altered until a hash that exceeds current difficulty is calculated. While the creators of SPECTRE state that 10 blocks per second is feasible using the protocol, HYCON will initially have a target of 1 block per second. Although the current prototype uses proof of work, we are very aware of the large amount of electricity being used to secure the bitcoin and ethereum networks, and are considering alternatives. One of the lesser-known methods that we are considering is Proof of Space[32], which requires miners to pre-compute and store large amounts of data, then search through the files for a solution that satisfies the current difficulty. This uses very little electricity and has been demonstrated to be effective by Burst coin and Space mint.

Mining Process in Detail

The initial stage of the mining process is the encoding and hashing of the contents of the block header, which will not change as a result of the mining process. These contents are the references to the previous blocks, the Merkle root of the transactions to be contained within the block, the block target difficulty, the block timestamp, and the root of the Merkle-Patricia Trie representing its current state after the transactions in this block(see Wallets & Accounts section for more information).

Block Header (for pre hash)
Previous Blocks: Array of 32 Byte Hashes
Merkle Root: 32 Byte Hash
Difficulty: 4 Bytes
Timestamp: 8 Bytes
State Root: 32 Byte Hash

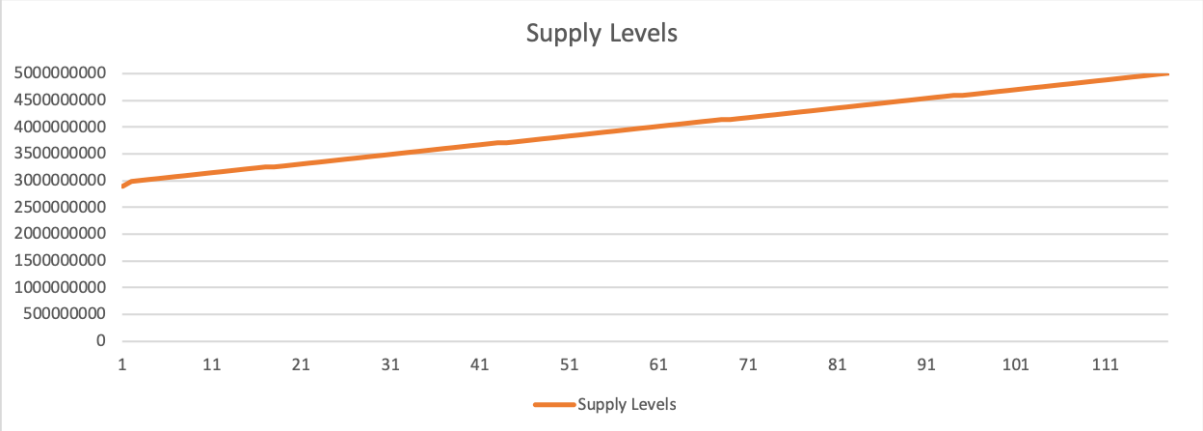
This data is hashed using the 64 Byte version of Blake 2b to provide an unchanging pre-hash to be furnished to a GPU or CPU miner. This step is necessary, particularly for GPU mining, as HYCON block headers have variable length due to multiple possible parent blocks. GPU mining software works best when furnished with a fixed length data structure so pre-hashing is necessary. The 64 byte header pre-hash is then combined with an 8 Byte nonce that is incremented by one for each hash attempt using the Cryptonight hash algorithm. The combined header pre-hash and nonce are hashed together to return a 32 byte hash representative of the block. This hash is then compared to the difficulty specified in the block header, and if the correct difficulty threshold is reached, the nonce is returned to be included in the finished block header and published.

Stratum Integration & XMRig

HYCON makes use of the Stratum protocol to allow for support of GPU mining using a modified version of XMRig[39].

Mining Rewards

On successful completion of the Proof of Work for a new block, the miner is rewarded with HYCON. The mining process for HYCON is planned to last a substantial amount of time. The reward for mining a block was initially set at 240 HYCON. This was then reduced to 120 HYCON after the GHOST protocol update, and down to its current value of 12 HYCON per block.



Wallets & Accounts

HYCON Wallet GUI

A full node running the HYCON software has access to a locally hosted web GUI allowing for wallet operations, transactions, and blockchain exploration. The GUI was written using React to allow for a lightweight high-performance interface.

HYCON Wallet

HYCON wallets employ industry standard elliptic curve cryptography methods for transaction signing, specifically secp256k1 [33], as well as implementing mnemonic codes for wallet recovery as specified by BIP 39 [40], to allow for ease of integration for third party wallet providers. Provision has also been made for HD (Hierarchical Deterministic) wallets as specified by BIPs 32 and 44. [41][42]

HYCON Addresses

HYCON addresses are generated as 20 byte arrays sliced from the 32 Byte Blake2b hash of the associated public key. For human readability the addresses are output as Base58 strings, prefixed with a capital H. The final 4 characters of the string act as a checksum for the address. The checksum is calculated in three steps. First, the 32 byte blake2b hash of the address is calculated. Then, this hash output is encoded to a Base58 string. Finally, the first 4 characters from this string are taken and appended to the string representation of the address. The use of a checksum in this manner minimizes the chance of mistyped addresses being accidentally used.

Accounts & Balances

In order to keep track of the spending and balances of HYCON participants, it is necessary to implement an accounting model. The model chosen for use in HYCON is based on the one used in Ethereum and described in the Ethereum Yellow paper[34], a data structure called a Merkle Patricia Trie[35]. Each block contains the world state as a result of the publication of its set of transactions. The value of which is the blake2b hash of the merkle patricia root representing the account data for all HYCON accounts.

The account data that is stored shows the balance of that particular account in HYCON. It references the most recent block of that particular account and a nonce value representing how many transactions that particular account has initiated. The nonce value is used to protect against replay attacks, and the previous block reference is an optimisation to allow for quicker transaction history querying as well as easier tracking for SPECTRE in the case of a double spend.

The blake2b hash is used in the accounting model as it allows for expedient hashing required to process numerous transactions and balances.

Synchronisation

HYCON will employ a headers first approach to initial synchronisation with the network. On the first startup, and subsequent startups thereafter, a message is sent to connected peers asking for a number of headers with a block height subsequent to a certain block height (the maximum block height currently stored in their local database). Upon receipt of those headers, the blocks are verified and if they are missing from the local database, the full block data is requested from a connected peer. The received blocks are verified again on receipt and then added to the database if they are validated. As blocks can only be added to the database once their parent blocks have been included this process is necessarily sequential.

CONCLUSIONS

This whitepaper's discussion began by exploring the limitations of existing cryptocurrencies, which was the basis for the entire Infinity Project. The vision of the Infinity Project is to provide a fast, secure, scalable, and user-centric blockchain and cryptocurrency ecosystem available for mass adoption. Through the combination of the SPECTRE protocol and Cryptonight and BLAKE-2b hashing algorithms, we have proposed a new cryptocurrency that is both safe and expedient. With the adoption of the methods elaborated herein, the HYCON cryptocurrency and Infinity Project offers a valuable and differentiated addition to the global cryptocurrency landscape.

REFERENCES

[1] Blake2.net. (2017). BLAKE2. [online] Available at: https://blake2.net/ [Accessed 16 Oct. 2017].

[2] CoinDesk. (2016). Understanding The DAO Attack - CoinDesk. [online] Available at: https://www.coindesk.com/understanding-dao-hack-journalists/ [Accessed 20 Nov. 2017].

[3] Croman, K., Decker, C., Eyal, I., Gencer, A.E., Juels, A., Kosba, A., Miller, A., Saxena, P., Shi, E., Sirer, E.G. and Song, D., 2016, February. On scaling decentralized blockchains. In International Conference on Financial Cryptography and Data Security (pp. 106-125). Springer Berlin Heidelberg.

[4] Decker, C. (2017). BitcoinStats. [online] Bitcoinstats.com. Available at: http://bitcoinstats.com/network/propagation/ [Accessed 10 Nov. 2017].

[5] Decker, C. and Wattenhofer, R., 2013, September. Information propagation in the bitcoin network. In Peer-to-Peer Computing (P2P), 2013 IEEE Thirteenth International Conference on (pp. 1-10). IEEE.

[6] digiconomist.net. (2017). Bitcoin Energy Consumption. [online] Available at: https://digiconomist.net/bitcoin-energy-consumption [Accessed 16 Nov. 2017].

[7] Digiconomist. (2017). *Ethereum Energy Consumption Index (beta) - Digiconomist*. [online] Available at: https://digiconomist.net/ethereum-energy-consumption [Accessed 8 Dec. 2017].

[8] The Economist. (2007). The end of the cash era. [online] Available at: http://www.economist.com/node/8702890 [Accessed 27 Sep. 2017].

[9] Ethereum Blog. (2014). Toward a 12-second Block Time - Ethereum Blog. [online] Available at: https://blog.ethereum.org/2014/07/11/toward-a-12-second-block-time/ [Accessed 27 Sep. 2017].

[10] Etherscan.io. (2017). Ethereum Average BlockSize Chart . [online] Available at: https://etherscan.io/chart/blocksize [Accessed 16 Nov. 2017].

[11] Ethstats.net. (2017). Ethereum Network Status. [online] Available at: https://ethstats.net/ [Accessed 16 Nov. 2017].

[12] Goland.org. (2017). How to make block chains strongly consistent – Stuff Yaron Finds Interesting. [online] Available at: http://www.goland.org/why_block_chains_are_strongly_consistent/ [Accessed 27 Sep. 2017].

[13] Goland.org. (2017). The block chain and the CAP Theorem – Stuff Yaron Finds Interesting. [online] Available at: http://www.goland.org/blockchain_and_cap/ [Accessed 27 Sep. 2017].

[14] Google Developers. (2017). Protocol Buffers | Google Developers. [online] Available at: https://developers.google.com/protocol-buffers/ [Accessed 20 Oct. 2017].

[15] James-Lubin, K. (2015). Blockchain scalability. [online] O'Reilly Media. Available at: https://www.oreilly.com/ideas/blockchain-scalability [Accessed 16 Nov. 2017].

[16] Koteska, B., Karafilovski, E. and Mishev, A. (2017), Blockchain Implementation Quality Challenges: A Literature Review : Proceedings of the SQAMIA 2017: 6th Workshop of Software Quality, Analysis, Monitoring, Improvement, and Applications, Belgrade, Serbia, 11-13.9.2017.

[17] Malanov.A, (2017). Six main disadvantages of Bitcoin and the blockchain. [online] Kaspersky.com. Available at: https://www.kaspersky.com/blog/bitcoin-blockchain-issues/18019/ [Accessed 16 Nov. 2017].

[18] Motherboard. (2017). One Bitcoin Transaction Now Uses as Much Energy as Your House in a Week. [online] Available at:

https://motherboard.vice.com/en_us/article/ywbbpm/bitcoin-mining-electricity-consumption-ethereum-energy-climate-change [Accessed 20 Nov. 2017].

[19] Nakamoto, S., 2008. Bitcoin: A peer-to-peer electronic cash system.

[20] The NodeSource Blog - Node.js Tutorials, Guides, and Updates. (2014). Why Asynchronous?. [online] Available at: http://nodesource.com/blog/why-asynchronous/ [Accessed 16 Nov. 2017].

[21] Park, J.H. and Park, J.H., (2017). Blockchain Security in Cloud Computing: Use Cases, Challenges, and Solutions. Symmetry, 9(8), p.164.

[22] Poon, J. and Dryja, T.. (2016). The Bitcoin Lightning.network [online] Available at: https://lightning.network/lightning-network-paper.pdf.

[23] Raiden-network.readthedocs.io. (2017). Raiden Specification — Raiden Network 0.2.0 documentation. [online] Available at: https://raiden-network.readthedocs.io/en/stable/spec.html [Accessed 7 Dec. 2017].

[24] Reitwiessner, C. (2017). zkSnarks in a Nutshell [online] Available at: http://chriseth.github.io/notes/articles/zksnarks/zksnarks.pdf [Accessed 23 Nov. 2017].

[25] Sirer, E.G. and Song, D., 2016, February. On scaling decentralized blockchains. In International Conference on Financial Cryptography and Data Security (pp. 106-125). Springer Berlin Heidelberg.

[26] Sompolinsky, Y., Lewenberg, Y. and Zohar, A., 2016. SPECTRE: A Fast and Scalable Cryptocurrency Protocol. IACR Cryptology ePrint Archive, 2016, p.1159.

[27] Sompolinsky, Y. and Zohar, A., 2015, January. Secure high-rate transaction processing in bitcoin. In International Conference on Financial Cryptography and Data Security (pp. 507-527). Springer, Berlin, Heidelberg.

[28] Son, M. (2017). Bitcoin's Rise Happened in Shadows of Finance. Now Banks Want In. [online] Bloomberg.com. Available at: https://www.bloomberg.com/news/articles/2017-10-05/bitcoin-s-rise-happened-in-shadows-of-finance-now-banks-want-in [Accessed 7 Dec. 2017].

[29] Swan, M., 2015. Blockchain: Blueprint for a new economy. " O'Reilly Media, Inc.".

[30] VISA (2017). Visa Inc. Facts & Figures . [online] Available at: https://usa.visa.com/dam/VCOM/global/about-visa/documents/visa-facts-figures-jan-2017.pdf [Accessed 20 Nov. 2017].

[31] Yli-Huumo, J., Ko, D., Choi, S., Park, S. and Smolander, K. (2016). Where Is Current Research on Blockchain Technology?—A Systematic Review. PLOS ONE, 11(10), p.e0163477.

[32] Dziembowski, Stefan; Faust, Sebastian; Kolmogorov, Vladimir; Pietrzak, Krzysztof (2015). "Proofs of Space". 9216: 585–605.
Available at: https://eprint.iacr.org/2013/796.pdf

[33] Secg.org. (2010). Standards For Efficient Cryptography 2, [online] Available at: http://www.secg.org/sec2-v2.pdf [Accessed 20 Jan. 2018].

[34] Wood, G., 2014. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum Project Yellow Paper*, *151*.

[35] Ethereum. (2018). *ethereum/wiki*. [online] Available at: https://github.com/ethereum/wiki/wiki/Patricia-Tree [Accessed 22 Jan. 2018].

[36] Team Hycon. (2018). *Team-Hycon/Genesis-View*. [online] Available at: https://github.com/Team-Hycon/Genesis-View [Accessed 22 Jan. 2018].

[37] Cryptonight Hash Function Cryptonote.org. (2018). [online] Available at: https://cryptonote.org/cns/cns008.txt [Accessed 2 Feb. 2018].

[38] JustCryptoNews. (2018). *Sia Coin - BitMain Antminer A3 Blake (2b) ASIC Miner Announced*. [online] Available at: https://www.justcryptonews.com/340/sia-coin-bitmain-antminer-a3-blake-2b-asic-miner-announced [Accessed 2 Feb. 2018].

[39] Monero (XMR) CPU Miner - GitHub. (2018). *xmrig/xmrig*. [online] Available at: https://github.com/xmrig/xmrig [Accessed 6 Feb. 2018].

[40] Palatinus, M., Rusnak, P., Voisine, A., Bowe, S.. *bitcoin/bips/bip-0039*. [online] Available at: https://github.com/bitcoin/bips/blob/master/bip-0039.mediawiki [Accessed 6 Feb. 2018].

[41] BIP32. (2012). *bitcoin/bips*. [online] Available at: https://github.com/bitcoin/bips/blob/master/bip-0032.mediawiki.

[42] BIP44 (2014). *bitcoin/bips*. [online] Available at: https://github.com/bitcoin/bips/blob/master/bip-0044.mediawiki.

[43] The Monero Project. (2018). Monero: A Scheduled Network Upgrade is Planned for April 6. [online] Available at: https://getmonero.org/2018/03/28/a-scheduled-protocol-upgrade-is-planned-for-April-6-2018-03-28.html.

APPENDICES

APPENDIX A - GENESIS BLOCK

Discussion

The HYCON genesis block was published on January 4th, 2018 at 3:15am KST(GMT+9) to coincide with the 9th anniversary of the publication of the Bitcoin Genesis Block. Below is a summary of the information contained within the block. The block, along with decoding software written in several programming languages, is available online at the Team Hycon github repository.

The block itself contains a header and six transactions. The six transactions represent the minting of the HYCON that is to be initially allocated. See Appendix B for full details of the token and budget allocation.

The block header contains the mining difficulty, the root of the merkle tree of the transactions, the root of the world account states, and the timestamp of the block.

The remainder of the block data is made up of the individual transactions which are loading up the accounts to be used for future token distribution. It was felt that encoding the transactions within the genesis block in this manner was the most transparent method for preparing for the wide release of HYCON, as it will allow all of the funds from these accounts to be tracked from the beginning.

Contents of HYCON Genesis Block

When decoded, the contents of the HYCON genesis block are as follows:

Block Header
Difficulty: 0 Merkle Root:
cff5f8a5381ce41e26bf3f5f7b658dcef0d4935dfd791460614feb894ff36457

State Root: e08408cb5bf38fb2652676af953d169c7997dd2af88299163b9a389b9d6a3ed4

Timestamp: Thu Jan 04 03:15:05 KST 2018

Transactions
ICO Account
Account Address(raw): 9565e92e694ef206abe21d65d3a93996682d41f7 Amount:
2,000,000,000 HYCON

Airdrop Account
Account address(raw): fa7042154efb88d06c198ef106ca31aed57e6875 Amount:
400,000,000 HYCON

Team Account
Account address(raw): 8bab45e2f5c79c00d539ae1a65dbd1f8fd416ca7 Amount:
500,000,000 HYCON

Development Fund
Account address(raw): 7a7b31e5aced4889a75d1042a6f1204d2a889af8 Amount:
500,000,000 HYCON

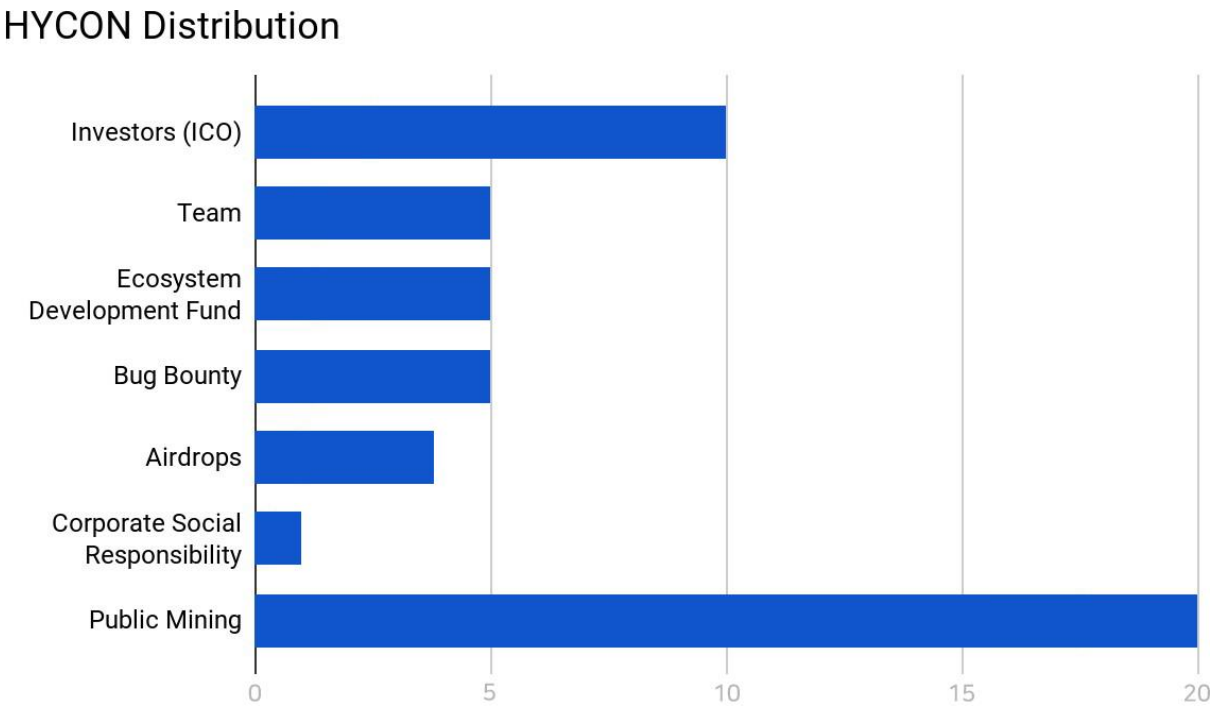
Bug Bounty
Account address(raw): 571a6e4554afbb09ee7da1ae20c18dbca9fabc46 Amount:
500,000,000 HYCON

Corporate Social Responsibility
Account address(raw): 11d8046e6cd88f9e580b84a0b10c7c452f0030fc Amount:
100,000,000 HYCON

APPENDIX B - Coin DISTRIBUTION & BUDGET ALLOCATION

Coin Distribution

The total available number of HYCON to be issued is 5 Billion. The method for allocation is as follows



As can be seen in the above chart, the majority of HYCON coins will be available through public mining, with 2 billion HYC allocated for that purpose. The remaining 3 billion HYC have been minted within the Genesis block and allocated to HYCON accounts responsible for the distribution of those coins once the network is live.

This 3 billion HYC have been allocated for six purposes. The largest portion is for the participants of the ICO and other pre-investors, with a total of one third of the initially minted coins (1 billion HYC) being allocated thus. 500,000,000 HYC will be provided to the team, to the Ecosystem Development fund, and to the Bug Bounty programs respectively. 100,000,000 HYC will be allocated towards the Corporate Social Responsibility of HYCON. The remaining 400,000,000 HYC are to be airdropped through events or other yet to be decided upon methods.

It should be noted that the Genesis block contains a 2 billion allocation for the ICO fund, however after consultation with the community, the number of tokens to be made available was halved, with 1 billion being available for that purpose.

Budget Allocation

The primary focus for the allocation of funds from our ICO is to facilitate the securing of talented development and securing the long term future of the project. As such, 70% of the funds raised will go into funding the future research and development of HYCON and the Infinity Project, including the Infinity platform and the Infinity decentralised exchange. It is important to note that while these are separate projects, the HYCON ICO is the only fund raising drive being taken to ensure the liquidity of the entire project moving forward.