0xBitcoin

(Whitepaper Revision 1.0.1)

0xBitcoin: The Decentralized Bitcoin Token for Ethereum

Abstract

The Ethereum Network has proven itself as the world’s first ecosystem for permissionless, transparent and immutable software applications. These software applications, typically taking the form of Smart Contracts, can all seamlessly interact with each other. To facilitate this process, various standard protocols have been developed such as the ERC20 standard for a common ‘token’ format so that these Smart Contracts can pass scarce, owned, and transferable data between one another without a centralized mediator. Up until 2018, every ERC20 token has been distributed in a matter that is generally known to align with ‘securities.’ The tokens are sold to ‘investors’ by the ‘creator’ under the pretenses that the ‘creator’ will perform some action to make the tokens more valuable. It should be clarified that Bitcoin is distributed via ‘bitcoin mining’ and therefore aligns itself as a ‘commodity’ and not a ‘security.’ This whitepaper will describe the first ERC20 token that aligns itself as a ‘commodity’ since it is distributed only using ‘Proof of Work Mining’ identical to the Bitcoin model. This token is also transferred on a blockchain in a method very similar to Bitcoin and so therefore interfaces with other software and with the world in a manner which is effectively identical to Bitcoin. This token has several advances that set it apart from Bitcoin such as the ability to directly interact with Ethereum Smart Contracts and the rest of the Ethereum Ecosytem in a permissionless way.

Background

0xBitcoin is the implementation of Bitcoin in Solidity and is the first decentralized ERC20 token for Ethereum. It is an open source community project, not led by an official team or corporation, and therefore does not have ICO capital or other vast amounts of currency/capital that a centralized token project would have. We believe as a community that decentralization is the true flavor of the blockchain and that is the architecture that provides open and transparent trust for users. We also believe that Ethereum and ERC20 tokens are a significant segment of the future of blockchain technology.

0xBitcoin is designed to be used as a decentralized ‘bitcoin-like’ token within the Ethereum ecosystem and beyond. It avoids problems related to centralization and security because it is powered by the Ethereum Network and by globally distributed anonymous miners. Since it follows a standard protocol (ERC20), it is stored in a traditional Ethereum wallet and it is transferred using standard software which supports EIP20/ERC20 tokens. Since every 0xBitcoin token has been mined in a completely decentralized manner, there is no central body or central organization which controls or enforces any aspect of 0xBitcoin. The community owns and operates the token in a flat structure and every individual has the same power over the smart contract as any other individual. This is on purpose in order to follow the same model of Bitcoin and to establish 0xBitcoin as a commodity.

One of the most effective side effects of Satoshi Nakamoto's desire to secure the original Bitcoin network with Proof of Work hash mining was tethering and bootstrapping the coin to computing power, thereby removing centralized actor jurisdiction. Transitioning the responsibility of work back onto individual miners, government organizations would have no jurisdiction, and indeed visibility, of mined 0xBitcoin. Government oversight is removed from an equation whereby miners are providing economic effort in direct exchange of a cryptographic commodity. This facilitates relatively decentralized distribution and establishes all involved parties as stakeholders. 0xBitcoin is a first in class token that allows projects to be funded not by centralized, direct-fiat conversion, but through decentralized computing power.

Name Origin of 0xBitcoin

The name 0xBitcoin is derived from a combination of the name of the decentralized and mined commodity Bitcoin with the term ‘0x’ which implies that the asset lives on the Ethereum Network. This is implied because all Ethereum addresses begin with the characters ‘0x.’ The 0xBitcoin contract is located at Ethereum address 0xb6ed7644c69416d67b522e20bc294a9a9b405b31 (https://etherscan.io/address/0xb6ed7644c69416d67b522e20bc294a9a9b405b31) and has validated transparent code which can be audited on the Etherscan service.

Ethereum and ICOs

The Ethereum blockchain in its current state exists as a thriving permissionless ecosystem which allows any individual to store immutable records in a permissionless, invulnerable and transparent manner. There is no other database system in the world that has this ability except for Ethereum and other similar blockchains. As blockchain applications become richer and more numerous, there is a need for alternative distribution models than the ICO. Indeed, there have been proposals to mitigate some initial investment risks through the recent introduction of the DAICO model (Cunningham, 2018) that rely on timed and automated value transfers via the DIACO smart contract tapping mechanism. However, this does not align a token smart contract as a non-security and still has the potential to put investors at risk if not implemented carefully. Allowing users of the network direct access to tokens by performing computations as a proof of work supplies allows any smart contract to distribute a token in a safe, slow, and controlled manner similar to the release of a new commodity.

As of 2018, all Ethereum token distribution methods are badly flawwed and able to be Sybil attacked. A Sybil attack is a form of computer security attack in which one human pretends to be many humans with multiple computer accounts in order to manipulate a system in a malicious way. All ICOs and airdrops are highly susceptible to Sybil Attacks and so there is no way to verify that all ERC20 tokens distributed in these manners have or have not been compromised. 0xBitcoin and its unique Proof of Work distribution method are resistant to Sybil attacks. This means that 0xBitcoin is the first trustless Ethereum token in the world.

Current and Proposed Use Cases

As an implementation of the original Bitcoin software as an Ethereum Smart Contract, 0xBitcoin (or 0xBTC) combines advantages from both Bitcoin and Ethereum. The asset is decentralized, permissionless, mined and scarce just like Bitcoin which means it shares all of Bitcoin’s usecases and properties as a transparent and permanent digital record of value. However, above Bitcoin, 0xBitcoin has the speed and scalability of the Ethereum network and is compatible with all ERC20 token services. This means it can be stored in any Ethereum wallet, is as secure as Ethereum, and can act as ‘the bitcoin’ for the Ethereum ecosystem. This is important because Bitcoin is not able to communicate with or interact with the Ethereum smart contract ecosystem. With 0xBitcoin, the Ethereum network is now effectively upgraded with the ability to interface with a commodity which shares all of the same properties as Bitcoin. Now, all Ethereum smart contracts can hold, transfer, and trade bitcoin-like tokens permissionlessly and can do so based on immutable rules set forth using their own computer code.

To elaborate, the commodity Ether is being used for many purposes within the Ethereum network. The ultimate usability of Ether as a decentralized store of value is unknown. This is because Ether is designed as a medium for securing the Ethereum network and not only as a form of 'bitcoin' for Ethereum. For example, if Proof of Stake is implemented for Ethereum, Ether will no longer be mined using Proof of Work. This will likely leave 0xBitcoin as the only mined asset on Ethereum. In this way and others, Ether may be transformed in such a manner as to make it best for securing the network and not as a good medium of exchange. This message has already been implied by the Ethereum development team in 2017. 0xBitcoin intends to help fulfill a role that Ether currently plays in the Ethereum network. 0xBitcoin intends to be the primary medium of exchange and store of value for the Ethereum network. This will allow Ether to fulfill its original intended function to secure the network at scale and to be the lifeblood of the Ethereum network.

The Decentralized Token

Since 0xBitcoin is mined like Bitcoin, it acts just like a commodity and as such its value has a relationship to the difficulty of 'mining' it using computation power. This is analogous to gold having a value related to the difficulty of finding and smelting gold ore. The current state of the Ethereum ICO market with its demonstrable failure rate leaves investors vulnerable to holding pseudo-value backed only by speculation. 0xBitcoin mitigates this problem by providing the Ethereum network with a decentralized bitcoin-like asset which is able to fill the role of a multitude of centralized tokens in a more invulnerable and trustless format.

This powerful mechanism frees individuals from having to use a third party exchange, susceptible to security holes and wallet compromise, and third party escrows. The movement away from centralization is a core tenant of what Satoshi Nakamoto originally intended with classic Bitcoin (Nakamoto, 2009). 0xBitcoin has the facilities to help keep the Ethereum ecosystem open, accountable, trustless and decentralized at every step in the value transfer process. Unlike Bitcoin, 0xBitcoin can interact decentralzied exchanges such as EtherDelta and ForkDelta since it is compatible with Ethereum smart contracts. This means that while Bitcoin can only be traded using centralized means, 0xBitcoin can be traded permissionlessly within immutable permanent smart contracts which are not able to be censored or restricted by central entities. This is another clear advantage and is closer to fulfilling Satoshi's complete vision.

Account System

As an ERC20 token, 0xBitcoin uses a traditional Ethereum account. These accounts are free and are impossible to hack or to steal from, given that the private key has not been exposed. 0xBitcoin can be stored in a Ledger Nano, Trezor or any other wallet that supports ERC20 tokens.

Mining

There have been mintable or mined tokens proposed for Ethereum in the past but none of them have ever successfully implemented Proof of Work or automated difficulty adjustment and so never became pure decentralized currencies. 0xBitcoin is mined using a simple Keccak256 (Sha3) algorithm using the following methodology:

```
   keccak256(nonce, minerEthAddress, challengeNumber) < difficultyTarget
```

The nonce is a random number selected by the mining software. The mining software mines to try to find a valid nonce. If the above statement evalutates to true, then the nonce is a valid solution to the proof of work. The challengeNumber is just a recent Ethereum block hash. Every round, the challengeNumber updates to the most recent Ethereum block hash so future works cannot be mined in the past. The miner's Ethereum Address is part of the hashed solution so that when a nonce solution is found, it is only valid for that particular miner and man in the middle attacks cannot occur. This also enables pool mining. The difficulty target becomes smaller and smaller automatically as more hashpower is added to the network.

Pool Mining

When mining 0xBitcoin, whenever a miner submits a solution, the miner must pay a small gas fee in order to execute the Ethereum smart contract code for the mint() function. If the gas fee is too low, the solution will take too long to be mined and if difficulty is not at equillibrium then another mint() solution from another miner will likely be mined first. This renders the original miners solution invalid and the transaction will revert(). To alleviate gas fees for miners, they can instead mine into a pool. This way, the pool will then submit the solutions to the smart contract and pay a gas fee. Then the pool will typically take a small percent of the rewards and give the rest to the miner for providing the PoW solution.

Since the miner's ethereum address is included in the proof of work, pools require that miners mine using the pool's ethereum address. This way, the miner cannot submit full solutions to the contract while only giving partial solutions to the pool. If the miner is mining on behalf of the pool (using the pools address in the PoW algorithm) then it will not be able to submit any of those solutions to the smart contract without a revert(). This allows pools to operate without being cheated by the miners.

Typically, a pool will accept 'partial solutions' from miners which means the miners will receive 'shares' from the pool for solutions that are close to valid but not quite valid. This follows the same methodology as Bitcoin and Ethereum Proof of Work pool mining. Probability theory states that, given enough close solutions, a full solution will eventually be found.

Smart Contract

Typically, ERC20 tokens will grant all tokens to the owner or will have an ICO which demands that amounts of Ether be sent to the owner for an initial offering of tokens. Instead of granting tokens to the 'contract owner', all 0xBitcoin tokens are locked within the smart contract initially. These tokens are dispensed, 50 at a time, by calling the function 'mint' and using Proof of Work, similar to mining bitcoin classic. The 0xBitcoin smart contract is the first token to adhere to the ERC541 Draft Specification. As such the following Smart Contract methods are explicitly supported:

Token

ERC-20 Interface

name

Returns the name of the token - e.g. "`0xBitcoin Token`".

OPTIONAL - This method can be used to improve usability,
but interfaces and other contracts MUST NOT expect these values to be present.

```
function name() constant returns (string name)
```

symbol

Returns the symbol of the token. e.g. `"0xBTC"`.

OPTIONAL - This method can be used to improve usability,
but interfaces and other contracts MUST NOT expect these values to be present.

```
function symbol() constant returns (string symbol)
```

totalSupply

Returns the total token supply.

```
function totalSupply() constant returns (uint256 totalSupply)
```

balanceOf

Returns the account balance of another account with address `_owner`.

```
function balanceOf(address _owner) constant returns (uint256 balance)
```

Mining Operations

mint

Returns a flag indicating a successful hash digest verification. In order to prevent MiTM attacks, it is recommended that the digest include a recent ethereum block hash and msg.sender's address. Once verified, the mint function calculates and delivers a mining reward to the sender and performs internal accounting operations on the contract's supply.

```
function mint(uint256 nonce, bytes32 challenge_digest) public returns (bool success)
```

Mint Event

Upon successful verification and reward the mint method dispatches a Mint Event indicating the reward address, the reward amount, the epoch count and newest challenge number.

```
event Mint(address indexed from, uint reward_amount, uint epochCount, bytes32 newChallengeNumber);
```

getChallengeNumber

Recent ethereum block hash, used to prevent pre-mining future blocks.

```
function getChallengeNumber() public constant returns (bytes32)
```

getMiningDifficulty

The number of digits that the digest of the PoW solution requires which typically auto adjusts during reward generation.Return the current reward amount. Depending on the algorithm, typically rewards are divided every reward era as tokens are mined to provide scarcity.

```
function getMiningDifficulty() public constant returns (uint)
```

getMiningReward

Return the current reward amount. Depending on the algorithm, typically rewards are divided every reward era as tokens are mined to provide scarcity.

```
function getMiningReward() public constant returns (uint)
```

Mining Debug Operations

getMintDigest

Returns a test digest using the same hashing scheme used when minting new tokens.

```
function getMintDigest(uint256 nonce, bytes32 challenge_digest, bytes32
challenge_number) public view returns (bytes32 digesttest)
```

OPTIONAL - This method can be used to improve usability,
but interfaces and other contracts MUST NOT expect these values to be present.

checkMintSolution

Verifies a sample solution using the same scheme as the mint method.

```
function checkMintSolution(uint256 nonce, bytes32 challenge_digest, bytes32
challenge_number, uint testTarget) public view returns (bool success)
```

OPTIONAL - This method can be used to improve usability,
but interfaces and other contracts MUST NOT expect these values to be present.

Minting New 0xBitcoins

The 0xBitcoin Token was deployed to the Ethereum blockchain in February, 2018, with the following attributes:

- No pre-mine
- No ICO
- 21,000,000 tokens total supply
- Difficulty target auto-adjusts with PoW hashrate
- Rewards decrease as more tokens are minted
- ERC20 compatibility

As such, the only way for a user to acquire 0xBitcoins is to mine them or purchase them from miners on decentralized exchanges. The mint function is responsible for verifying the validity of the hash solution, updating the contracts internal state and issuing new 0xBitcoins.

```
function mint(uint256 nonce, bytes32 challenge_digest) public returns (bool
success) {
    uint reward_amount = getMiningReward();

    bytes32 digest =  keccak256(challengeNumber, msg.sender, nonce );

    if (digest != challenge_digest) revert();

    //the digest must be smaller than the target
    if(uint256(digest) > miningTarget) revert();

    uint hashFound = rewardHashesFound[digest];
    rewardHashesFound[digest] = epochCount;
    if(hashFound != 0) revert();  //prevent the same answer from awarding twice

    balances[msg.sender] = balances[msg.sender].add(reward_amount);

    tokensMinted = tokensMinted.add(reward_amount);

    //set readonly diagnostics data
    lastRewardTo = msg.sender;
    lastRewardAmount = reward_amount;
    lastRewardEthBlockNumber = block.number;

    //start a new round of mining with a new 'challengeNumber'
     _startNewMiningEpoch();

    Mint(msg.sender, reward_amount, epochCount, challengeNumber );

    return true;
}
```

figure 1. 0xBitcoin Smart Contract mint() function

The mining reward is initially gathered and follows the same algorithm as Bitcoin classic. Essentially following the paradigm of a fully decentralized monetary system, whereby the tokens are created by the nodes of a peer to peer network. The 0xbitcoin algorithm defines how the token will be created and at what rate.

As with Bitcoin, 0xBitcoins are generated every time a user discovers a new block by being the first to submit Proof of Work for each round. The rate of the block creation is adjusted every 1024 to aim for a relatively constant adjustment period equal to approximately 6 0xBitcoin blocks per hour. The number of 0xBitcoins generated per block is set to decrease logarithmically, having a 50% reduction every time half of the remaining supply has been mined. This ensures that the number of 0xBitcoins in existence will never exceed 21 million.

A unique 'nonce' has to be passed into the mint function along with the hash solution digest in order for tokens to be dispensed. To find this special number, it is necessary to run a mining program. More specifically, the PoW includes a recent Ethereum block hash combined with the wallet sender's address in order to prevent man in the middle attacks when minting new coins. The challenge and nonce are validated in solidity using the keccak256 hashing algorithm to decipher the challenge's digest. Once the digest has been extracted, it is validated to match the expected challenge result and then check to ensure that it is smaller than the mining target difficulty.

The mining reward is calculated based on the logarithmic halving algorithm making the 0xBitcoin token a reliably deflationary asset. The award is immediately assigned to the sender's wallet address and the ‘tokens minted count’ is incremented within the smart contract for any other software to monitor. Notably, the contract then validates that the tokens minted count is less than or equal to the maximum supply or the given halving era that transaction is taking place. Next, the contract records diagnostics reflecting reward address, amount and ether block number for the purpose of public transparency and for other software to monitor.

Difficulty Calculation and Adjustment

After every block is minted, the smart contract will determine if it is time to adjust the difficulty. This occurs every 1024 mined blocks. Just before this occurs, the contract increments the reward era if necessary - this is, if the tokens minted count has exceeded the maximum era supply which is calculated via a simple halving algorithm:

max_era_supply = total_supply - (total_supply / (2 * (reward_era + 1)))

This means that the first era supply is 10500000 tokens, the second era supply is 15750000 tokens, the third era supply is 18375000 tokens and so forth. During the first era, the block reward for a mint() is 50 tokens. During the second era, the reward is 25 tokens. During the third era, the reward is 12.5 tokens and so forth. There are forty eras total until the mining will halt. This is expected to take about 100 years at which time 0xBitcoin can be used as a decentralized digital currency for Ethereum.

The reward era is used to calculate the mining reward. Next, the 0xBitcoin smart contract adjusts the difficulty by first determining how many Ethereum blocks had been mined since the last adjustment. If less than 1024*60 Ethereum blocks had been mined, 0xBitcoin is being mined too quickly and the difficulty will increase. This is accomplished by reducing the size of the ‘target’. When the target is smaller, valid nonces for minting are more rare and are harder to find for future mining rounds. Alternatively if 0xBitcoin is being mined too slowly the target will increase in value in order to make minting more easy to accomplish. All difficulty targets are bound within minimum and maximum difficulties of 216 and 2234 respectively.

Calculating Mining Hashrate

To calculate approximate hashrate or approximate time to find a solution, the following equation can be used:

```
TimeToSolveBlock (seconds) = (difficulty * 2 ^ 22) / hashrate (hashes per
second)
```

Risks and Challenges

0xBitcoin is implemented as an Ethereum ERC20 token and so its success is largely dependent on the success of the Ethereum Network. If Ethereum cannot scale using methods such as Plasma, Casper, and the Loom network, then 0xBitcoin will not be able to realize its full potential as the fastest and most effective decentralized currency in the world.

Frequently Asked Questions

Does 0xBitcoin have its own Blockchain?

No. 0xBitcoin exists on the Ethereum Blockchain as a Smart Contract. This allows it to leverage a faster, more secure and modern crypto environment.

Why are there times when a lot of mints get reverted?

The difficulty was too low compared to hashrate and so multiple valid solutions were submitted to the contract in a very short amount of time. Only one can be accepted each round and so the others are reverted.

How does pool mining work with 0xBitcoin?

Essentially the same way that pool mining works for classic Bitcoin, except 0xBitcoin pools must pay gas fees to the Ethereum network.

How often does difficulty update?

Every 1024 blocks.

How does the difficulty update?

It increases up to 100% or down 50% with fractional changes in between in an effort to be approximately 60x slower than eth block rate, or roughly 10 minutes.

Will there be a reward halvening event and when?

At 10.5m tokens mined and when half the remaining has been mined then half of that remaining then half of that remaining, up to 40 iterations.

Since 0xBitcoin is Proof of Work doesn't that mean it is bad for the environment?

As long as cryptocurrencies exists, mining will always exist. Even though mining expends energy, it ultimately reduces corruption in society by providing humanity with decentralized and transparent transactional ledgers. Therefore the idea similar to humanity having to pay for a gigantic decentralized accounting system or police network which is reducing the widespread financial corruption across the globe. Just as we pay police officers and accountants for their service, we pay blockchain for its service in the form of energy and computation.

Whitepaper Contributors

1. Infernal_toast (contract deployer)
2. Jay Logelin (jlogelin@fas.harvard.edu)

References

Satoshi Nakamoto. Bitcoin: A Peer-to-Peer Electronic Cash System, 2009.
http://www.bitcoin.org/bitcoin.pdf.

Logelin J and 0xBitcoin communitiy members. ERC 541 - Mineable Token Standard Draft, 2018.
https://github.com/ethereum/EIPs/pull/918

Fabian Vogelsteller and Vitalik Buterin. ERC-20 Token Standard, 2015. URL https://github.com/ethereum/EIPs/blob/master/EIPS/eip-20-token-standard.md.

TrustNodes. The First PoW Bitcoin Like Token Launches on Ethereum, February 16, 2018. https://www.trustnodes.com/2018/02/16/first-pow-bitcoin-like-token-launches-ethereum

Vitalik Buterin. Ethereum White Paper, 2014. https://github.com/ethereum/wiki/wiki/White-Paper

Epstien J. Why Proof of Work in Bitcoin Means Proof of Value in the Real World, December 20, 2017. https://www.neverstopmarketing.com/proof-work-bitcoin-means-proof-value-real-world/

Bitfury Group Limited. "Proof of Stake versus Proof of Work", 2015. http://bitfury.com/content/5-white-papers-research/pos-vs-pow-1.0.2.pdf

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

Dai W. "b-money", 1998. http://www.weidai.com/bmoney.txt

Back A. "Hashcash - a denial of service counter-measure", 2002. http://www.hashcash.org/papers/hashcash.pdf

Cunningham A, Ethereum Co-Founder Announces DAICO, a new ICO Fundraising Model (January 15, 2018). https://discover.coinsquare.io/investing/daico-new-ico-fundraising-model/