tellor

A Decentralized Oracle - Whitepaper

Abstract

Smart contracts on Ethereum are fully self contained and any access to off-chain data is restricted. By creating a system where inputs to a data series are secured by a network of staked miners, the “Tellor Oracle” creates trustless access to off-chain information. This paper highlights the structure of this system and gives an in-depth overview regarding the incentives and assumptions used to ensure an honest input of data to the oracle.

Table of Contents

I. Introduction

Smart contracts on Ethereum are fully self-contained and any information or access to off-chain data is restricted. Tellor solves this problem by creating a system where parties can request the value of an off-chain data point (e.g. ETH/USD) and miners compete to add this value to an on-chain data bank, accessible by all Ethereum smart contracts. Inputs to a data series are secured by a network of staked miners. The main Tellor smart contract creates a time series of each requested data series and aims to become the standard source of high value data for decentralized applications. This oracle, “Tellor”, utilizes similar incentive mechanisms to other cryptocurrency systems through the issuance of Tellor’s token, Tributes, that are used to request a specific data series from miners.

II. Background

One of the biggest problems limiting decentralized finance growth is the lack of a secure and decentralized price feed. Tellor aims to enable the growth of the DeFi industry by providing a secure source of for high quality pricing data.

Blockchain networks, the Ethereum network specifically, allow for fast and secure transfer and creation of digital goods in addition to the storage and execution of tamperproof programs that can manage digital assets.[1] These programs, once deployed on-chain, cannot be changed. They are available to everyone with access to the chain, will execute based on the defined parameters and interactions (transactions), and are verified by the blockchain’s consensus mechanism. These characteristics allow anonymous parties to enter into binding digital agreements, or smart contracts. However, because of the redundancy built into the consensus mechanisms of these networks, there is no native vehicle for reading off-chain data (e.g. internet API’s). If a smart contract relies on off-chain data to evaluate or execute a function, parties current rely on these options:

- *Manually feed the data to the contract*—the data can be easily compromised and is not a trustless mechanism
- *Trust a centralized party to provide the data*—efficient, but not trustless
- *Rely on a group of trusted known parties (Proof-of-Authority consensus)*—not trustless[2]
- *Schelling Point systems (incentivize users to provide data and reach consensus)*—unnecessary for simple price data, can lead to long waiting times to reach consensus, and confidence on the data can erode if there is a conflict of interest[3]

[1] www.ethereum.org

[2] MakerDAO’s DAI uses a set of 15 known parties to provide data to their DAI contracts.

[3] https://www.lesswrong.com/posts/yJfBzcDL9fBHJfZ6P/nash-equilibria-and-schelling-points; Some projects that have tried to incentivize their users to provide data are Augur, Gnosis, Aeternity

None of these strategies have proven to be optimal: efficient, trustless and decentralized. Unfortunately, for smart contracts to bring true utility, off-chain data is necessary.

The **Tellor Oracle** provides an efficient, trustless and decentralized alternative for off-chain data. It provides the infrastructure for decentralized applications to query off-chain data by properly incentivizing miners to provide data.

III. The Tellor Oracle

The Tellor Oracle is an on-chain data bank where miners compete to add the data points. To create a properly incentivized system Tellor mints a native token, “Tributes.” Parties pay Tellor Tributes to submit a request for data to the Oracle. Based upon the reward assigned to each request, the Oracle selects the best funded query every ten minutes to create a challenge for miners to solve[4]. Each query collects specific data (e.g. ETH/USD or BTC/USD prices) and makes it available on-chain. The Tellor Tributes are used to pay fees to miners, who add the official data point, and is used to secure the network via a staking requirement for miners and a data validation voting system. Five submissions are necessary to determine the official data point. Created with a similar structure as 0xBitcoin[5], the Tellor Oracle uses a mineable proof-of-work (PoW) token, but along with the PoW solution, miners are also required to provide an off-chain data point. The first five miners to provide the PoW solution and off-chain data point are rewarded with newly minted tokens and the accumulated payout for the specific data request**.**   Proof-of-work has proven to be the gold standard for crypto economic consensus mechanisms and Tellor utilizes it within a hybrid model to secure the oracle. In addition to the security provided by the PoW process, we have added an additional layer of security through the deposit of Tributes which miners are required to stake before they are allowed to mine and risk losing their stake if their submitted values are successfully challenged.

A. Implementation

The Tellor Oracle utilizes a delegate proxy structure for its contracts, deploying two smart contracts: 1)TellorMaster.sol allows delegate calls from Tellor.sol to allow for data storage. TellorMaster holds the historically mined values that contracts can read from. 2)Tellor.sol holds and distributes the token supply, informs miners which values to submit, and has a built-in methodology for challenges. It provides the miners with necessary fields for data collection, allows miners to submit the proof and

[4] Similar to transaction fees in Bitcoin and Ethereum.

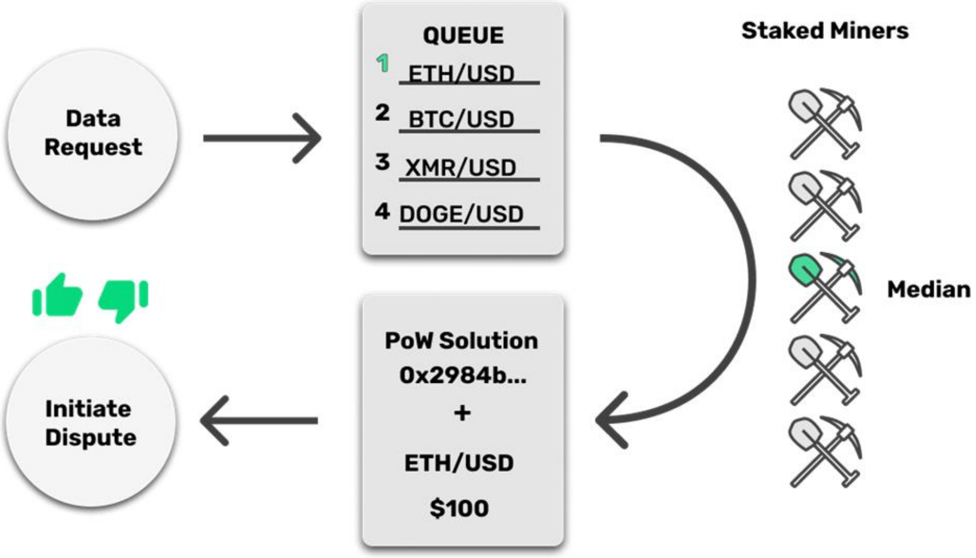
[5] https://0xbitcoin.org/

off-chain data, sorts the values, and allows users to retrieve the values and bid on which data series is mined next. The contract targets for new values to be mined every 10 minutes via difficulty adjustments. Which data series is mined is determined by which series has the greatest "tip" going to the miners.

The basic flow for adding and retrieving data is as follows:

1. The user submits a query to the Oracle using Tributes to incentivize miners to choose this query over other submissions.
2. Other users who want the same data pay or 'tip' this data series to further incentivize selection by miners.
3. Every 10 minutes, the Oracle selects the best funded query and provides a new challenge for miners to solve.
4. Miners submit their PoW solution and off-chain data point to the Oracle contract. The Oracle contract sorts the values as they come in and as soon as five values are received, the official value (median of the five) is selected and saved on-chain. The miners are then allocated their payout (base reward and tips).
5. Anyone holding Tellor Tributes can dispute the validity of a mined value within one day of it being mined by paying a dispute fee. Over the next week, Tellor token holders will vote on the validity of the data point; if the data point is deemed to be false, the miner will lose their stake. However, if the vote determines the value is correct, the reporting party's dispute fee is given to the reported miner.

Figure 1: Tellor Oracle



The official value appended to the timeseries is determined by a decentralized mechanism where five values are collected before the winning value is selected. The first five values received are sorted as they are submitted, all miners are rewarded equally (5 Tributes + tips) and median value will become the 'official' value. Once validated and processed the value is available for on-chain contracts to use.

Five miners was chosen as a result of looking at the success of malicious actors on Bitcoin and Ethereum, which can be measured by how many blocks have been reorganized in the system (creating malicious or competing chains). To explain, if a malicious miner attacks Bitcoin and mines 3 blocks in a row, the next block will still not build on top of it. Honest miners will instead 'reorg' or reorganize the blockchain by 3 blocks and take out the blocks mined by the bad actor. In this sense, one can never break Bitcoin; you can only break it for a certain amount of time.

In Bitcoin, the longest chain reorg was four.[6] In Ethereum it was 7, but the short block times make it much less severe. In our system, attacking our chain with 5 miners is similar to breaking a traditional system for 3 blocks; maliciously getting 3 out of 5 is sufficient to enter a malicious value. In the 10 years of Bitcoin, 3 or 4 length reorgs happened 6 times. Tellor however has additional security even above the longest chain security mechanism due to our staking requirement.

The data collection for the Tellor system is decentralized since mining, and by extension data submission, is open to everyone who stakes. Using the median value instead of the average (or simply just one) protects the value from being manipulated by a single party submitting an extreme value.

During the time that the value is being confirmed (one day), parties can challenge this submission. The challenge and data value are put up to vote by Tribute holders. This is described in detail in the Mining and Security section.

B. Incentives

Two types of incentives are implemented in this hybrid model, 1) rewards for PoW submissions and 2) structural incentives to promote accurate value submissions.

Miners are given two types of rewards:

1) A base reward per every successful submission
2) Tips given to miners to incentivize the selection of a query

[6] https://bitcoin.stackexchange.com/questions/3343/what-is-the-longest-blockchain-fork-that-has-been-orphaned-to-date/4638

Base reward

Similar to the way Ethereum rewards 'Uncles', or miners who were close to winning, the first five miners to submit a PoW and off-chain value are awarded the five Tributes each.

Tellor took into consideration two main problems when structuring payouts for our reward mechanisms:

- Race Conditions
- Mirroring

Race conditions occur when a user can see a pending transaction on Ethereum and then submit their own transaction with a higher gas price to front run this transaction. The main problem with race conditions in Tellor is that users will waste all of the potential mining reward on gas to outbid one another. When this happens, the security of our system goes down because as the expected return (mining reward minus gas costs) decreases, so does the PoW difficulty of our system.

Mirroring is when parties copy the previous submission's value. So if you have 5 miners submitting the price of BTC, a properly functioning system does not feature everyone just copying the first miner's submission. The goal of the five separate miners is for each miner to independently check the value and report it. When mirroring occurs, there is only one miner who is actually checking the value, thus reducing your n number of miners securing your system to one.

Based on Tellor's research, the following conclusions can be made: [7]

- Higher payout for median value encourages race conditions and mirroring
- Rewarding the median leads to bad values
- When all miners are rewarded equally race conditions are reduced, but mirroring remains
- Mirroring occurs when bad actors are absent

The official value will still be the median value but all miners will receive equal rewards. If Tellor solely takes the first submitted value, the cost of attack for POW would decrease. Race conditions will still occur but with less severity than when the median value is paid a higher reward. In addition, since we anticipate mirroring to be a problem and the only way to prevent it is to actually submit fake values on occasion to keep the system honest we plan to police the environment by using utilizing our devshare to submit malicious values if a sufficient number are not present.

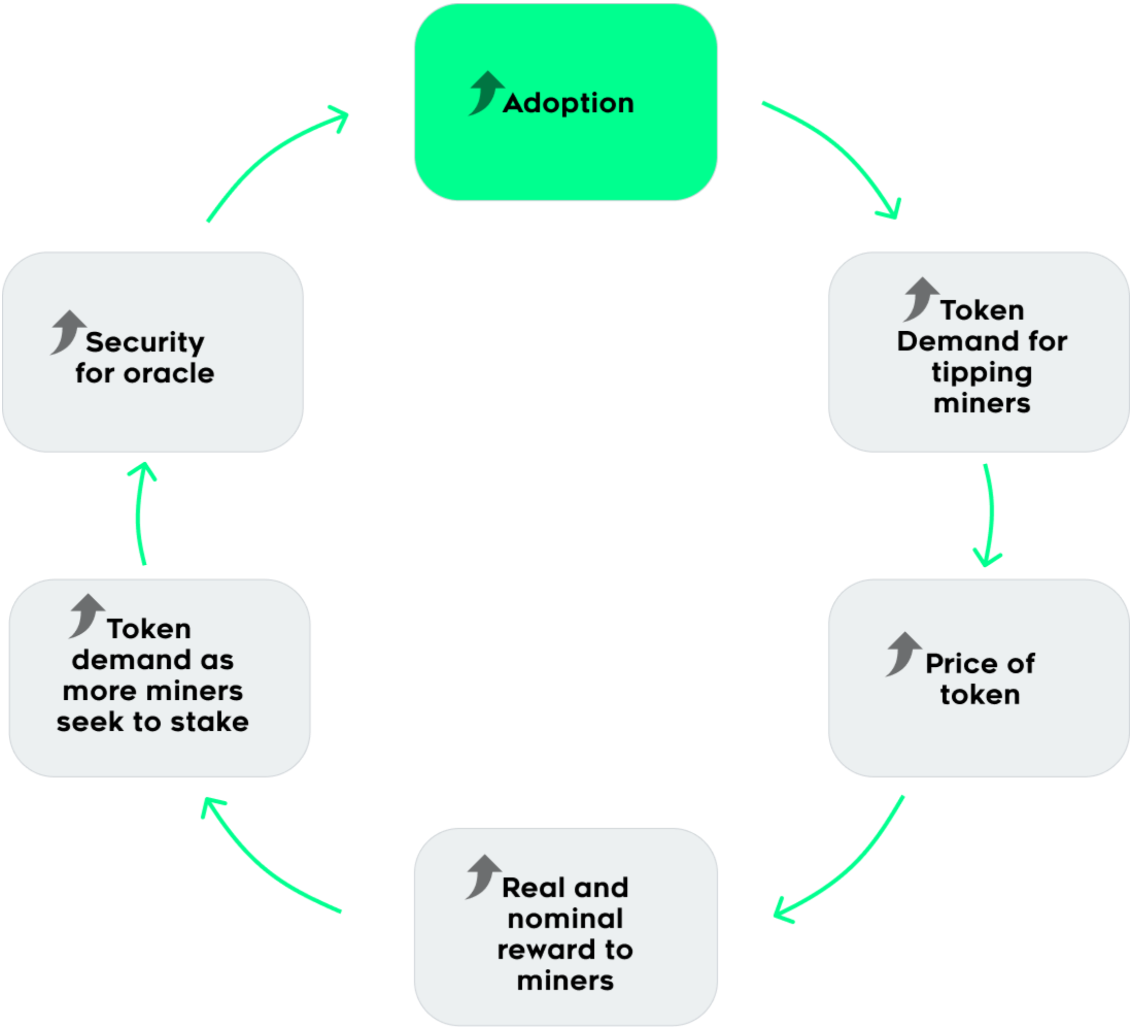
[7] https://medium.com/@tellor/tellor-security-and-rewards-1e365e1be8ae

Tips

Users incentivize miners to retrieve their value by posting a bounty to ensure the query they are interested on is mined. Akin to paying a higher gas fee for a prioritized transaction, this is a tip to the miners and is paid out evenly to the five miners. As the ecosystem expands, securing data to finalize or execute a contract will lead to higher tips and higher incentivization of miners to continue to mine.

Since the time target of the oracle is 10 minutes, there are only 144 queries per day on average. As the Tellor oracle is adopted, the queue will fill, and price competitions will take place. This is a self-fulfilling cycle as adoption increases so does demand, miner rewards, security and ergo further adoption.

Figure 3: Tellor Oracle Adoption Cycle


Adoption
Token Demand for tipping miners
Price of token
Real and nominal reward to miners
Token demand as more miners seek to stake
Security for oracle

Incentives to submit proper value

- Every miner is required to stake 1000 tokens

Miners must stake 1000 Tributes to be able to mine. Proof-of-stake allows for economic penalties to miners submitting incorrect values.

- Every accepted value can be challenged and put to vote by any Tellor Tribute token holder

Blockchains are secured via multiple avenues. The first is in the random selection process provided by PoW. The second is that even if the first security measure fails, the blockchain can create forks and different chains until the honest miners win out. Since our oracle system does not have the ability to fork easily, Tellor implements a finality period of one day after original data submissions by allowing parties to challenge data submissions.

C. Mining and Security

Mining

Miners are given the following information from the Tellor oracle contract

- Current Challenge
- Request ID
- Difficulty
- API Query String
- Tips associated with request

The Algorithm

One of the main challenges for a mineable token or any process that relies on mining is the surplus of solo ASICS currently available since if they are used on a small ecosystem, these specialized systems can quickly monopolize it. Tellor's proof of work challenge is designed to be different than the Bitcoin mining challenge. This setup requires miners to invest a significant amount of time to update the mining algorithm and should disincentivize malicious miners from becoming part of the ecosystem too early, allowing Tellor to grow and mature before larger players join and potentially dominate the hashpower.

The code to determine a successful mine for a given challenge and difficulty is:

```
current challenge = keccak256( nonce, current challenge, blockhash( block.number - 1);
solution = sha256( ripemd160( keccak256( current challenge, msg.sender, nonce);
new difficulty = difficulty +  difficulty * ( time target - ( now - time of last new value))/100;
```

The difficulty adjustment is based on the difference between the time target (10 minutes in Tellor) and the time it took to solve the previous challenge. For an example, if the five PoW challenges are submitted in 9 minutes, the difficulty will increase by 10% on the next challenge.

Submission

Miners submitting values must submit the following in order to be a valid submission:

- Successful PoW solution (nonce)
- Request ID
- Value of Requested Data

Security

Security is achieved through the Tellor Oracle's architecture (mining algorithm and selection process for median value) and incentives implemented for miners to promptly submit the correct values (see the "Incentives" section for further details).   Ultimate security however is provided by the Proof-of-Stake dispute resolution. Since any value that is disputed will be put to a vote by all token holders, the simple cost to break is:

$$\mathit{Token\ Holder\ Voting\ Share}\ \times \mathit{Price\ of\ Tributes}$$

This PoW/PoS hybrid model allows for Tellor to take advantage of the efficiency and minimalism of a pure PoW design as well as the final security of PoS. The main problem with PoW consensus mechanisms is that 51% attacks for significant time periods are relatively trivial on smaller chains. The problem with a pure PoS mechanism is that stakers are not properly incentivized to mine (since usually economic punishments are needed) and the general security of negative reinforcement properties do not promote competition in speed and accuracy. Both of these issues are solved through Tellor's hybrid model and the security of the Oracle should suffice for relatively large purposes shortly after launch.

Looking at our formula, we can summarize that security increases when:

- The share of token holders voting PoS disputes increases
- The price of the token increases
- Demand for the Oracle increases (tips)

The Dispute Process

Tellor implements a finality period of one day after original data submissions. This allows for any party to challenge data submissions of any of the five miners for up to one day after the blocktime of when the value is placed on chain.

A challenger must submit a dispute fee to each challenge.[8] Once a challenge is submitted, the potentially malicious miner who submitted the value is placed in a locked state for the duration of the vote. For the next week, tribute holders vote on the validity of the mined value. All Tribute holders have an incentive to maintain an honest Oracle and can vote on the dispute. A proper submission is one that corresponds to a valid query of requested within the time period between the release of the challenge and the submission of the value.

If found guilty, the malicious miner's stake goes to the reporter; otherwise the fee paid by the reporter is given to the wrongly accused miner.

Requested Data descriptions

Tellors system allows for purely on-chain requests (complete API query strings), but will also work over time to build out robust, standardized series based upon request ID. To give an example, a party can request the BTC/USD price as 'json(https://api.gdax.com/products/BTC-USD/ticker).price' or the Tellor system can build into its miner and community that the BTC/USD price corresponding to a given request ID is actually identified as something more robust such as a moving average of five different exchanges. Since the validity of a data point is ultimately decided by a vote on the Tellor system, a request ID can correspond to any number of different hard coded or even manual inputs to prevent a data provider from censoring Tellor miners (i.e. shutting off their API feed).

[8] https://medium.com/@tellor/staking-disputes-and-voting-ad09c66eb7bc

D. Adoption

A rule in distributed systems is that a system is only as decentralized as its least decentralized feature. Therefore, any smart contracts or protocols relying on oracles that claim decentralization need a purely decentralized oracle to maintain the integrity of their system. Tellor is building partnerships and working on commitments of use from currently deployed applications and working with these partners to ensure a smooth technical transition to Tellor. On the availability front, Tellor's dev-share and oracle payout structure will provide incentives to early miners to ensure a robust and secure system.

The Tellor team is working hard to ensure ease of use and availability.

To read data, for smart contracts that are currently using a centralized service, the update will be familiar and simple for testing and on mainnet. These are the steps users need to take:

1. npm install tellor
2. On their contracts use "is usingTellor" to access Tellor user functions

For full documentation, please go to: : https://tellor.readthedocs.io/en/latest/

Or utilize the open-source code on Github: https://github.com/tellor-io/TellorCore

IV. Tellor Tributes

The Tellor Tribute, the native token of the Tellor Oracle, incentivizes miners to provide data, keeps the oracles secure and allows DApps to request and access to on-chain data. The following sections provide an overview of the expected Tribute supply, current price constraints, and uses for the dev share.

A. Supply

The total supply of Tellor is determined by usage and mining rates. For the maximum supply, Tellor's supply will grow at the rate of the base reward * 144 queries per day. The graph below shows the Tellor supply and the growth rate assuming full utilization:

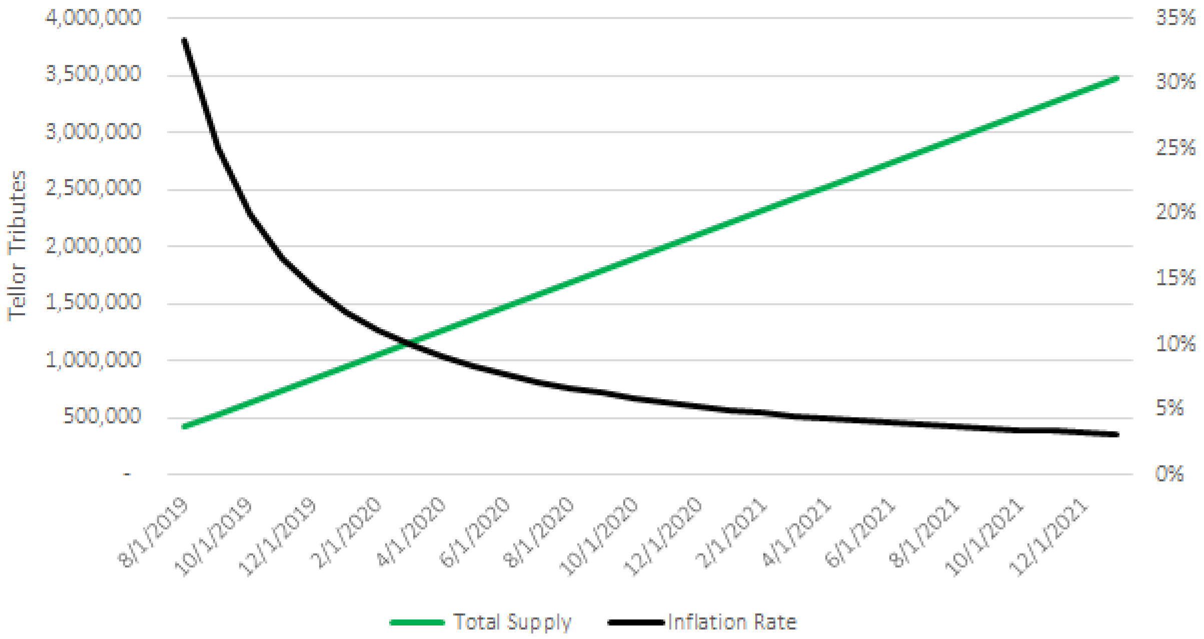


Chart 1: Expected Tellor's annual supply and growth rate (Assuming August 2019 Launch)

B. Price Constraints

Tellor's main competitors are centralized oracle services that can charge low fees for API data retrieval. Tellor is a premium oracle service which provides 144 API queries per day to the applications needing a secure oracle.

The system has several supply and demand constraints affecting the price:

- The security of the system is highly correlated with the price of the token and any monopoly or erosion of confidence in the system will decrease this security and hence the price.
- As the token price increases, less tokens are needed for each query to the miners (assuming demand in USD is stable). This will free supply and decrease the price of the token to an equilibrium level.
- As the price of the token rises, this creates increased security in the Oracle system by further incentivizing miners through the base reward.

C. Dev Share

The Tellor Oracle implements a ten percent developers share (dev share). This revenue stream will be managed by the Tellor team and utilized in the following ways:

- Ensure community participation and accurate voting for disputes
- Create and distribute efficient mining software
- Market and Promote the Tellor Oracle to ensure adoption which leads to greater mining incentives
- Create developer tools for utilizing the Tellor Oracle in production deployments
- Fund research and improvements to the oracle

Tellor will use the initial tokens to provide liquidity to users of the oracle. There is no pre-mine or token offering. Tokens will be sold on an as needed basis by Tellor to provide liquidity to the users and miners of the system. If you want to partner with us to utilize the oracle in your smart contracts, please contact us for details.

V. Potential Applications

Within the context of Ethereum, oracles can be thought of as authoritative sources of off-chain data. These data points allow smart contracts to receive and condition executional instructions. The biggest use case for off-chain data has been stablecoins, which utilize the ETH/USD price to maintain a peg.

As Tellor is a contract mechanism that allows oracle data to be derived in a competitive, decentralized manner - we envision a wide array of use cases for this product. Namely:

1. **Exchange-rate data:** interval-based exchange-rate values may be used to create trustless financial contracts
2. **Cross-Chain Information:** gathering the number of blocks or current block number on a given chain
3. **Static/pseudo-static data:** logging and indexing various identifiers, country codes, currency codes
4. **Damage verification:** what were the net total results in damage for insurance contracts
5. **Pseudorandom number generation:** to select a winner in a distributed-lottery smart contract, etc.

VI. Future

The Tellor team is already looking toward furthering research in the field of decentralized Oracles. Several solutions have already been identified as potential ways to increase security and speed of the Tellor Oracle.

- Zero-knowledge submissions

Zero-knowledge proofs (ZKP) allow for parties to prove that they know a value without revealing the value.[9] For the Oracle, parties could submit a ZKP for the mining solution along a hidden query value. Once the five parties are chosen as successful miners, they are then required to post the unhidden value (Oracle query) which corresponds to the hidden value submitted with the ZKP. This could help us save gas costs for submissions, prevent mirroring, and further incentivize miners to compete for the median value.

- TLS Notary Proofs

TLS Notary Proofs give assurances that a website was queried accurately and that no error was returned.[10] The Tellor Oracle has plans to utilize different levels of assurances that can be returned (or stored) with the query to ensure that miners are accurately reporting data from the requested query.

- Optimistic Implementation

The implementation of a complementing non-mining oracle system that allows for data submission by any party for the data requests. This complementary system assumes data submitters have the best intentions. This “optimistic” approach can allow for disputes by requiring a PoS and/or can be based on submitter reputation. This implementation would be considered less secure and would cater to projects/Dapps/users that may not be time sensitive and can “shop” around for data.

- Automatic Reporting and monitoring

Off-chain analysis for detecting outliers and reporting these to “gain” the “bad” miner’s stake. For example, reporting a value/miner, if the mean differs from median by certain amount.

[9] https://en.wikipedia.org/wiki/Zero-knowledge_proof

[10] https://tlsnotary.org/

VII. Conclusion

The Tellor Oracle is already developed and functioning on the Ethereum Rinkeby testnet and is available on Github at https://github.com/tellor-io/TellorCore

The Tellor Oracle provides a decentralized option for high value off-chain data. Tellor plans to deploy our current contracts and to continue research on creating a secure, scalable, and on-demand Oracle to help smart contracts achieve their true potential. By creating an oracle schema that uses an incented construct to derive the validity of off-chain data, we:

- **Reduce** the risks associated with single-party oracle providers, who can cut access to API data, censor certain users, or manipulate reported values for private gain.
- **Build** the foundation for a superior oracle system where data is derived from a distributed set of participants which have both an economic interest and a stake in the validity and success of the oracle data.
- **Create** an effective, secure, and incentivized system for off-chain data which disincentives dispersion and adversarial submissions.

If you are interested in using the Tellor Oracle, contributing to its development, or becoming a miner, please contact us at info@tellor.io.

APPENDIX 1 - SECURITY CONSIDERATIONS

Minimum Cost to Attack Analysis

The cost to successfully attack and break the Tellor oracle is determined by several factors:

1. The price of the oracle Tellor Tributes
2. Average price (in fiat) per query (Demand)
3. Stake amount for mining
4. Voting share of honest parties

where:

P = Price of Tellor Tributes
D = Average price (in fiat) per query (Demand)
S = Stake amount in tokens
V = Voting share of honest parties(in tributes)

Assuming that miners will only mine up to the reward amount minus a needed premium (assume 10%). The cost of a 51% attack on Tellor:

$$Cost\ to\ 51\%\ attack = Miner\ reward$$

where 5 is the per miner reward and

$$per\ Query\ Reward\ to\ winning\ miner = 5\ \times P + D$$

Additionally, to 51% attack the network, you would need to gain all ⅗ mining rewards to ensure you capture the median value

$$Cost\ to\ 51\%\ attack = 3\ \times (5P + D)$$

Each miner would also be required to stake tokens (S) in order to mine so:

$$Cost\ to\ 51\%\ attack = 3\ (S + 5P + D)$$

This simple analysis though fails to account for the fact that invalid values will never be accepted if disputed. Therefore, one would additionally need to corrupt the vote.

$$Cost\ to\ corrupt\ Tellor\ Token\ Vote = V\ \times P$$

So therefore the total cost to break the Tellor system is:

$$V\ \times P + (S + 5P + D)$$