

oSnap

How to verify oSnap Proposals

What is the ASSERT_TRUTH identifier?

oSnap uses the `ASSERT_TRUTH` identifier can be found [here](#). `ASSERT_TRUTH` is intended to be used as a default price identifier for UMA's [Optimistic Oracle V3 contract](#) that allows asserters to make claims about the state of the world.

Price settlement can happen only in one of two ways:

- Return the 1 value if the claim is true.
- Return the 0 value if the claim is false or cannot be resolved.
-

Part 1: Snapshot Proposal Created

When a Snapshot proposal is created, a Discord ticket is created with a Tenderly simulation of the transaction payload. Here is an example:

?

The main objective of verifying the Snapshot proposal before voting is to ensure:

- The transaction payload can be executed
- Comparing the intent of the proposal against the transaction payload
- The transaction payload doesn't have any unintended consequences or malicious intent
-

In the Tenderly simulation, if the `Status` value is `notSuccess`, that is an indicator the transaction payload is unable to be executed and requires further review.

The input shows the transactions array that will be executed if the Snapshot proposal passes:

?

Verifying the transaction data

1. Go to <https://lab.miguelmota.com/ethereum-input-data-decoder/example/>.
2. .
3. Input the contract ABI into the ABI section
4.
 1. Take the 'to' value from the transaction object and input it into etherscan (or the block explorer based on the chainID. For etherscan, you would go to <https://etherscan.io/address/{to}>
- 5.
6.
 1. }.
7.
 1. Click the 'contract' tab
8.
 1. Scroll down to the 'Contract ABI' section and copy everything
9. 6.
 1. Paste it into the ABI section
- 10.

Here is what the example would look like:

?

The 'Output' for a Transfer should look as the below:

...

```
Copy { "method": "transfer", "types": [ "address", "uint256" ], "inputs": [ "9040e41eF5E8b281535a96D9a48aCb8cfaBD9a48", { "type": "BigNumber", "hex": "0x14adf4b7320334b9000000" } ], "names": [ "to", "amount" ] }
```

...

Verify the following:

- For any BigNumber values, go to <https://playground.ethers.org/>
- and input `BigNumber.from("{INPUT_VALUE}").toString()`
- For example, `BigNumber.from("0x14adf4b7320334b9000000").toString()`
- would be `25000000000000000000000000`
- To scale this value, go to the token contract address and check decimals. For the ACX token, the decimals are 18 so the scaled value is `25000000`
- You can use <https://eth-converter.com/>
- if the decimals are 18.
- Confirm that the above value is mentioned in the actual Snapshot proposal.
- ?
-

Part 2: oSnap Transactions Proposed

After a Snapshot proposal is completed, if the proposal is valid the transactions are proposed which creates another Discord ticket. [Here](#) is an example thread in the UMA Discord.

The first objective is to verify the oSnap rules against Snapshot Proposal. Click the Snapshot space URL, in the above example that is <https://snapshot.org/#/acrossprotocol.eth>. Go through the list of proposals and confirm the explanation from the Discord message matches the Snapshot space.

Validators should do the following to verify the oSnap proposal:

- Check the Snapshot space referenced in the rules string.
- Check that the IPFS hash passed as the explanation lines up with an actual vote in that Snapshot space. They should be able to cross-reference the value with a link from a specific Snapshot proposal.
- Check the Snapshot proposal tied to the IPFS explanation value met the minimum voting period and quorum.
- Check that the Snapshot proposal passed according to the Snapshot strategy.
-

Compare Snapshot IPFS Transaction data against Proposed Transactions

- ?

- Then copy the 'Transaction Hash':

?

?

[illegible]

11

- 'data' from Tenderly matches the transaction data from Snapshot.
- 'value' from Tenderly matches the transaction data from Snapshot.
- 'to' from Tenderly matches the transaction data from Snapshot.
- Verify the explanation:
 - Go to <https://playground.ethers.org/>
 - and input the following explanation to ethers.utils.toUtf8String("f^{explanation}
 - }).toString()
 - Take the returned value and input it into the following url <https://snapshot.4everland.link/ipfs/fexplanation>
 - } and verify that the data matches from Snapshot

[Previous Polymarket](#) [Next Across](#) Last updated 1 month ago On this page Was this helpful? [Edit on GitHub](#)