Drand

Drand, pronounced dee-rand, is a distributed randomness beacon daemon written in Golang.

This page covers how Drand is used within the Filecoin network. For more information on Drand generallytake a look at the project’s documentation.

Randomness outputs

By polling the appropriate endpoint, a Filecoin node will get back a Drand value formatted as follows:

```

Copy { "round":367,
"signature":"b62dd642e939191af1f9e15bef0f0b0e9562a5f570a12a231864afe468377e2a6424a92ccfc34ef1471cbd58c37c6b020cf75ce9446d2aa1252a090250b2b1441f8a2a0d22208dcc09332eaa0143c
"previous_signature":"afc545efb57f591dbdf833c339b3369f569566a93e49578db46b6586299422483b7a2d595814046e2847494b401650a0050981e716e531b6f4b620909c2bf1476fd82cf788a110becbc7
}
```

- signature
- : the threshold BLS signature on the previous signature value and the current round number round.
- previous_signature
- : the threshold BLS signature from the previous Drand round.
- round
- : the index of randomness in the sequence of all random values produced by this Drand network.
-

The message signed is the concatenation of the round number treated as a uint64 and the previous signature. At the moment, Drand uses BLS signatures on the BLS12-381 curve with the latest v7 RFC of hash-to-curve, and the signature is made over G1.

Polling the network

Filecoin nodes fetch the Drand entry from the distribution network of the selected Drand network.

Drand distributes randomness using multiple distribution channels such as HTTP servers, S3 buckets, gossiping, etc. Simply put, the Drand nodes themselves will not be directly accessible by consumers; rather, highly-available relays will be set up to serve Drand values over these distribution channels.

On initialization, Filecoin initializes a Drand client with chain info that contains the following information:

- Period: the period of time between each Drand randomness generation.
- GenesisTime: at which the first round in the Drand randomness chain is created.
- PublicKey: the public key to verify randomness.
- GenesisSeed: the seed that has been used for creating the first randomness.
-

It is possible to simply store the hash of this chain info and to retrieve the contents from the Drand distribution network as well on the/info endpoint.

Thereafter, the Filecoin client can call Drand’s endpoints:

- /public/latest
- to get the latest randomness value produced by the beacon.
- /public/
- to get the randomness value produced by the beacon at a given round.
-

Using Drand

Drand is used as a randomness beacon for leader election in Filecoin. While Drand returns multiple values with every call to the beacon (see above), Filecoin blocks need only store a subset of these in order to track a full Drand chain. This information can then be mixed with on-chain data for use in Filecoin.

Edge cases and outages

Any Drand beacon outage will effectively halt Filecoin block production. Given that new randomness is not produced, Filecoin miners cannot generate new blocks. Specifically, any call to the Drand network for a new randomness entry during an outage should be blocked in Filecoin.

After a beacon downtime, Drand nodes will work to quickly catch up to the current round. In this way, the above time-to-round mapping in Drand used by Filecoin remains invariant after this catch-up following downtime.

While Filecoin miners were not able to mine during the Drand outage, they will quickly be able to run leader election thereafter, given a rapid production of Drand values. We call this acatch-up period.

During the catch-up period, Filecoin nodes will backdate their blocks in order to continue using the same time-to-round mapping to determine which Drand round should be integrated according to the time. Miners can then choose to publish their null blocks for the outage period, including the appropriate Drand entries throughout the blocks, per the time-to-round mapping. Or, as is more likely, try to craft valid blocks that might have been created during the outage.

Based on the level of decentralization of the Filecoin network, we expect to see varying levels of miner collaboration during this period. This is because there are two incentives at play: trying to mine valid blocks during the outage to collect block rewards and not falling behind a heavier chain being mined by a majority of miners who may or may not have ignored a portion of these blocks.

In any event, a heavier chain will emerge after the catch-up period and mining can resume as normal.

Previous Consensus Next Proofs

Last updated6 months ago