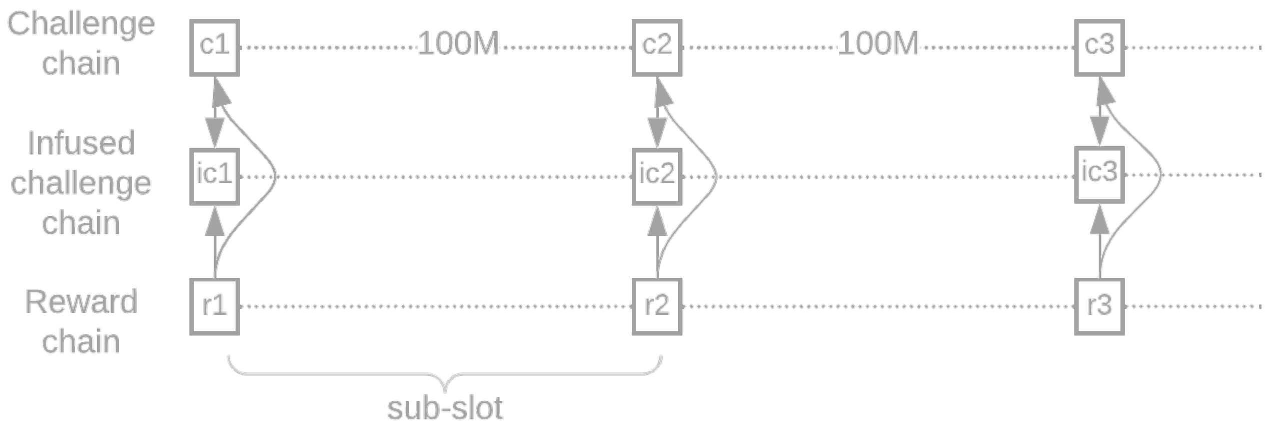
Challenges

The BPX consensus algorithm relies on timelords running VDFs for periods of time called *sub-slots*, which are adjusted periodically (and automatically) to take around 5 minutes (300 seconds). During every sub-slot, challenges are released by timelords, and a sort of mini lottery starts, where farmers check their plots for proofs of space. When farmers find a proof of space that qualifies, they broadcast it to the network.

The difficulty adjusts automatically to target 32 winning proofs for the entire network in each sub-slot, or about one winner every 9.38 seconds on average (32 winners per 300 seconds). The winning proofs are infused into the VDF at different times within the sub-slot.

> A sub-slot is always targeted to last 5 minutes. There is also a period of time called a *slot*. Typically, a slot and a sub-slot are exactly the same thing. However, in order to prevent long-range attacks, slots are required to have at least 16 blocks (and sub-slots are not). If a sub-slot ends with fewer than 16 blocks having been created, the same slot must continue for another sub-slot.

The consensus requires farmers to follow the heaviest chain, which is the chain that has the highest accumulated difficulty (usually the chain with the most blocks).



We can see three challenge points, c1, c2, and c3. At the these points timelords create challenges (256-bit hashes) which are provided as input to VDFs. Timelords take these hashes, and start computing a VDF on this challenge, for the specified number of iterations. In this example, each slot is 100,000,000 iterations. When the VDF is finished, the timelord publishes the new challenge and the proof of the VDF. An infusion of end-of-slot information happens at the end of each sub-slot.

A challenge is always a 256-bit hash. The base info that is always included in this hash is the challenge chain VDF. However, the infused challenge chain, SubEpochSummary, difficulty, and sub slot iters might also be included, depending on where we are in the epoch cycle:

```
class ChallengeChainSubSlot(Streamable):
    challenge_chain_end_of_slot_vdf: VDFInfo
    infused_challenge_chain_sub_slot_hash: Optional[bytes32]  # Only at the end of a slot
    subepoch_summary_hash: Optional[bytes32]  # Only once per sub-epoch, and one sub-epoch delayed
    new_sub_slot_iters: Optional[uint64]  # Only at the end of epoch, sub-epoch, and slot
    new_difficulty: Optional[uint64]  # Only at the end of epoch, sub-epoch, and slot
```

Sub-slot: a segment of a fixed number of VDF iterations, subject to periodic work difficulty adjustments, which automatically target a time of 5 minutes.

Sub-slot iterations: determines how many VDF iterations each sub-slot must have. This number is periodically adjusted.

Challenge: a sha256 output string. It is used as a proof-of-space challenge for farmers' plots. It is also used for the challenge chain VDF, and is sometimes referred to as a *challenge hash*.

As you can see on the image above, there are three VDFs being executed concurrently, each of which serves a different purpose. In the networking protocol, the three VDF proofs are usually passed around together, in what is called an *end of sub-slot bundle*.

Revision #6
Created 5 June 2023 14:41:23 by Admin
Updated 30 October 2024 18:13:58 by Admin