Why I think trusting that rollup full nodes will give light clients valid block headers is the best solution

Reason 1: PoW and PoS light clients also rely on the p2p network to sync

In PoW I could publish a bunch of blocks that appear to have the right amount of work performed but the hash doesn't check out. In PoS I could publish a bunch of blocks where the signatures don't check out. In both cases the light client would none the less have to download each header and verify that the hash or signatures don't match before discarding the block header as invalid and knowing which is the canonical chain.

The reason this attack doesn't work in practice is that in PoS and PoW the nodes verify the signatures and hashes before propagating them. Therefore these invalid blocks never get distributed throughout the network and are "forgotten".

We don't have this luxury in Celestia because these invalid blocks get posted to the main-chain regardless of their validity and will be "remembered" by the main network. Hence why we need new mechanisms to combat this attack.

Note however, that these invalid blocks are "forgotten" by the rollup peer to peer network in the same way that the PoW and PoS peer to peer networks forget invalid blocks by not propagating them.

Trusting the rollup sub-network full nodes to only give you the valid block headers while syncing your light client seems equivalent in that you are assuming the sub-network only "remembers" valid blocks. There may be some formal equivalence there which implies that the trust assumptions in the rollup context are not any worse than in standard PoW and PoS light clients.

Reason 2: You probably need to get historical fraud proofs from an honest node anyway

I can imagine 2 scenarios:

1. find the historical fraud proofs on the Celestia main chain
2. trust honest full nodes to share the historical fraud proofs as they sync

For 1 to work the rollup aggregator would have to post a fraud proof on the Celestia main-chain for each and every invalid block at the previous height. This only worsens the effects of the spam attack, forcing more and more data to be posted on chain.

In the case of 2, you are essentially trusting that there's an honest full node who will give you all the fraud proofs you need to tell which chain is valid. This is the same assumption you would be making if you just downloaded the block headers directly from a full node.

Hence 1 is impractical and 2 requires the same trust assumption as just downloading the block headers directly from an honest full node. Therefore you may as well just trust that there is an honest full node who will give you all the valid block headers and bypass this whole process.

Reason 3: It would be easy to recover from syncing with a dishonest full node

So long as the light client is connected to the wider p2p network i.e. is not being eclipse attacked, then it should receive messages showing that the rest of the network is building on a different chain. This could alert the light client to connect to other full nodes to check if they are on the wrong chain. A single fraud proof from an honest full node would be enough to discard the false chain and re-sync to the right chain.

I believe this vulnerability to eclipse attacks is the same in light clients of PoW and PoS chains. As @musalbas likes to point out, all blockchains need to assume that you can connect to at least 1 honest node as a peer for anything to work.