

collisions, but makes header format verbose. Presently, it is included in the draft Eleutherus/GP header format (see below) as “suggested”. As with anything else, feedback and suggestions are welcome. There is also a brief section on the broader subject of protocol “chattiness” in the context of blockchain networking.

Moving on, ChainID also acts as a unique identifier for any given Eleutherus/GP blockchain network. Again, something that the Bitcoin protocol and other “tall” blockchain protocols do not do, nor need to. So, the genesis user will have to select a unique proof-of-work operation for the new network. Now that is one of the few requirements that Eleutherus has in place.

This sort of flexibility to choose any proof-of-work operation to establish a network is a boon and not a bane. In fact, the user can choose to deploy a proof-of-work operation that is already being used by another blockchain network. As this could create confusion due to the lack of consistency, there needs to be a unique identifier and that’s where the ChainID comes in. Like any other field in the header, the ChainID is calculated by the mining software before it begins calculations for the proof-of-work operations. Therefore, modifying it without invalidating the proof-of-work (same as any other header field) is next to impossible.

3. Unstructured P2P networking → enables the use of different P2P overlay networks

On some levels, this is arguably the simplest of the three. Conceptually, switching between one Eleutherus and another is entirely similar to switching between Bitcoin’s MainNet and TestNet (or SegNet, etc...). However, there are some key factors that you must take note of.

To begin with, the new network is going to be quite different, particularly, its proof-of-work operation and historical blockchain. Also, other aspects like the type of applications supported, network ownership, and operational settings might differ. Still, switching from one P2P overlay to another is not in itself problematic. Bitcoin already contains a workable solution that can be implemented by calling the -testnet flag or including testnet=1 in the bitcoin.conf file.

This can also be done manually by setting the default listen port, the RPC connection port, etc... Thus, for now, it works as well as it needs to. However, there are some good reasons as to why this is an area for development. Primarily, usability and secondarily, the discovery issue, as mentioned above.