DendrETH Specification

DendrETH is a superset of ALC. Thus, we do not define again the `LCU` algorithm, but merely the proposed improvements. We define two data structures to use in our protocol. First, the slashing evidence, or `evidence`. Secondly, we define the slashing action $\mathcal{S}$. A slashing starts with a beacon state $\mathcal{B}$ [68] and a slashing action. We represent the action of slashing a set of validators $\mathcal{N}$ by $\mathcal{S} \xrightarrow{\texttt{evidence},\mathcal{N}} \{1\}$

We define `evidence` as having several attributes:

1) `BlockHeader`*: an attested block header, which is the contender that originates the slashing.
2) $\mathcal{C}_{i+1}$: the next sync committee.
3) $\pi_{\mathcal{C}_{i+1}}$: the $next\_sync\_committee\_branch$ is the Merkle path that authenticates the next sync committee.
4) `BlockHeader`: the finalized block header.
5) $\pi_{\texttt{BlockHeader}^*}$: a Merkle branch validating the finalized block header.
6) `aggregate`: the aggregated signature $\sigma_{N_{[1:512]}}$ of the nodes participating in the sync committee, $sync\_committee\_signature$ and the bitmap of the participants $bmap$.
7) $slot$: the signature slot.
8) $K_P^{N_1}, \ldots, K_P^{N_{512}}$: the sync committee public keys.
9) $root$: represents the root hash of a block that the evidence claims to be finalized.
10) π_{root}: Merkle proof showing the inclusion of $root$ in the state tree.

It is worth noting that there are two evidences provided in a slashing. The purpose of having two distinct shreds of evidence is to provide proof that a particular validator (or set of validators) committed an equivocation or another malicious act. Equivocation essentially means producing multiple conflicting pieces of information for the same context.

The slashing action $\mathcal{S}$ has:

1) $\mathcal{N}$: a list of validators to be slashed.
2) `evidence`.
3) `BlockHeader`$_{|HeaderChain|}.root$: the recent finalized block header root.
4) $slot$: recent finalized slot.

The complete slashing algorithm has two parts: `IdentifySlashing` and `EnforceSlashing`. The latter should only be called if the former returns 1 (otherwise, it means that the slashing evidence is not valid).

Algorithm 2: `Slashing` algorithm in DendrETH - `IdenfitySlashing`

Input: $\mathcal{B}, \mathcal{S}$, `evidence`[2]
Output: $\{\perp, 1\}$

slash $\leftarrow 0$ ▷ `slash flag`
assert $\mathcal{S}.slot >$ `evidence`$.slot$
▷ `asserts both evidences are sequential`
if $evidence[0].slot =$ $evidence[1].slot \wedge evidence[0].BlockHeader^* =$ $evidence[1].BlockHeader^*$ **then**
  slash $\leftarrow 1$
end if
if $evidence[0].\mathcal{C}_{i+1} \neq evidence[1].\mathcal{C}_{i+1}$ **then**
  slash $\leftarrow 1$
end if
`linear`$[0] \leftarrow$ `evidence`$[0].root ==$ `evidence`$[0]$.`BlockHeader`$.root$
if $\neg(final(evidence[0]) \vee final(evidence[1]))$ **then**
  assert `BlockHeader`$_{|HeaderChain|}.root = \emptyset$
end if
▷ `Checks to prevent slashing validators who signed an alternate history non-maliciously`
`canonical_is_0` $\leftarrow$ `evidence`$[0]$.`BlockHeader`$.slot \geq$ `evidence`$[1]$.`BlockHeader`$.slot \wedge slot ==$ `evidence`$[0]$.`BlockHeader`$.slot \wedge$ `BlockHeader`$_{|HeaderChain|}.root ==$ `evidence`$[0].root \wedge$ `linear`$[0]$
▷ `Might check` $\mathcal{B}$ `and` $\mathcal{S}$
if $canonical\_is\_0 \wedge slash$ **then**
  return 1
end if

Description of Algorithms 2 and 3 is in the text

Algorithm 3: `Slashing` algorithm in DendrETH - `EnforceSlashing`

Input: $\mathcal{B}, \mathcal{S}$, `evidence`[2]
Output: $\{\perp, 1\}$

`ToSlash` $\leftarrow \emptyset$
for *key* ***in*** $\mathcal{S}.evidence[0].K_P^{N_1}, \ldots, K_P^{N_{512}}$ **do**
  `ToSlash` $=$ `ToSlash` $\cup$ `GetPublicKey`$(key,$ `aggregate`$.bmap)$
end for
for *val* ***in*** $\mathcal{S}.\mathcal{N}$ **do**
  ▷ `asserts at least one validator` val `to be slashed` $\in \mathcal{B}.validators$
end for
assert `ValidateSlashingEvidence`(`evidence`$[0]$, $\mathcal{S}$.`BlockHeader`$_{|HeaderChain|}.root, slot,$ $\mathcal{B}$.`GenesisRoot`)
$\mathcal{S} \xrightarrow{\texttt{evidence},\mathcal{N}} \{1\}$

Fig. 5: DendrETH specification (`slashing` algorithm)