@JustinDrake @vbuterin

Suppose the proposers and validators for a shard have been bribed by an attacker.

They no longer publicly

sign off on or propose collation bodies or headers within their shard.

At the end of an epoch, the validators share with other shards a new merkle root for a set of transactions, Tx, that have been unwitnessed by honest members a shard.

Since no block Tx was signed and given to members of the shard, there can be no fraud proof against the shard for its actions. In fact, we don't even know whether Tx is valid or not.

To help solve this problem, we propose the following proof of visibility scheme:

It uses:

(1) Proof of Custody Justin's post

In proof of custody a user may present a ZK proof that:

D = \textrm{SHA3}((Tx[0]\oplus P) \hspace{1mm}\oplus\hspace{1mm} ... \hspace{1mm}\oplus\hspace{1mm} \textrm{SHA3}(Tx[n] \oplus P))

.

We can modify this so that the ZK proof is also based on public randomness, R, which is changed each epoch and the user's secret key S. We also use a commitment to the hash of Tx, H.

Present a ZK proof that:

D = \textrm{SHA3}((R\oplus Tx[0]\oplus S) \hspace{1mm}\oplus\hspace{1mm} ... \hspace{1mm}\oplus\hspace{1mm} \textrm{SHA3}(Tx[n] \oplus S))

.

S

is the secret key of some P

.

Tx or its hash is signed by the validators

\mathrm{Hash[Tx] =H}

Note: the randomness is generated after \mathrm{Hash[Tx] =H}

is committed and broadcast to the network/other shards.

This is to ensure that the attacker can't just change Tx for a given public and secret key (P, S) until a valid proof is found.

We also require that the digest D has 3 zeros as its least significant figures.

Now, with all this anyone presenting such a ZK proof will be one in a thousand. By that we mean, for an attacker to acquire such a proof, he must have bribed 1000 participants on average, presenting a TX block to all of them. (A participant will not know whether they can create the proof before they have Tx, randomness R, and have combined it with their private key S).

Giving their private keys S to the attacker for him to compute the digest is a highly risky endeavour, so the participants will not do so.

Whereas an attacker must have spread his funds over 1000 members, to generate a proof of visibility, we can instead issue a high block reward to the first person who produces proof of invalidity in Tx. Thus, a person will be more highly incentivised to report a faulty Tx or multiple Txs than take the attacker's bribe (by a factor of 1000 to 1 assuming equal funds).

Under the honest minority assumption, it is also likely the case that a member of the community disseminates Tx to the wider public for scrutiny even without financial incentivisation.

Modifications of this scheme may also be useful for aggregating votes efficiently, and may even help with super-quadratic sharding.

NB: This scheme should be less computationally intensive than presenting a zero knowledge proof that:

(1) All the transactions in Tx are valid

(2) The transactions in Tx are related to a merkle root M

And note: given a proof of visibility, some honest node with high probability has Tx, from which anyone can compute a valid Merkle root M’, and compare that with the M broadcast by validators