The first SUAVE code release with SGX in it is out.Sirrah: Speedrunning a TEE Coprocessor | Flashbots.

This is a great time to discuss a bunch more topics, now that we have a whole concrete example running in an enclave that's easy to modify.

- The Validator Collusion

problem mentioned in the post is for me the most surprising issue, which I think deserves more awareness since it does apply to every existing TEE-based smart contract blockchain. A quorum of validators could collude to decrypt all the data. So this isn't any better

as a trust model than MPC… just tidier and more efficient. But we could

do better if validators ran TEEs as well, using TEEs as an Anti-Collusion mechanism. So this is interesting to discuss.

- Expanding on the key manager.

This is the simplest possible key manager, it would be interesting to make Solidity contracts that more resemble the features from Oasis or Phala. They have a whole "council of gatekeepers" that multisig vote on mrenclaves. Things like notice periods are interesting too. It's just Solidity, anyone could jump in here

- Side channels.

I really want to see what this looks like running in SGX-Step to see what kind of memory access pattern it leaks. Lmk if interested in helping

- Bailing out of the TEE.

How could you take one of these applications and pass the confidential data to an External TEE application, like with an entirely different enclave? You shouldn't need to modify any precompiles to do this, just use the same DCAP attestation or something. Just a subversive idea