Hi Secret Community,

This thread is for kick starting a discussion for the tradeoffs around compartmentalizing.

As you may know, I was part of a team(sgx.fail) that did whitehat security vulnerability research on SGX applications, and in particular we found that Secret's master decryption key could be through xAPIC vulnerability, after working together on the disclosure the attack window was closed up (secret's post).

It's possible that something like this could happen again (let's hope not, but it's tough betting against it). Even among TEE-based smart contracts, Secret's architecture presents a really large attack surface. So we should consider hardening along the lines of what Phala and Oasis do, based compartmentalization (ekiden research paper).

In Secret, anyone can join the network with an enclave, i.e., there is no stake requirement or voting/approval requirement, the amount of nodes that could potentially be attacked using a future SGX vulnerability is quite large. Obscuro takes the same approach. In contrast, Oasis and Phala limit the access to key material to staked/bonded nodes — and even then, only on a contract-by-contract basis, with the master keys stored by an even smaller number of trusted (and eventually secret-shared) key manager nodes.

On the other hand, limiting who gets access to the key material has several disadvantages. It means taking a hit in decentralization, as running a node will be permissioned. You'll either need to have enough stake to be a validator or to be whitelisted. It reduces competition and diversity, and could potentially lead to a reduction in the performance of the network due to a smaller number of nodes providing services to dApps. In addition, any attempt to build in a mechanism which 'shards' the keys across contracts/validators, would potentially affect the network on several levels. Composability would be a big concern (i.e., contracts calling contracts), liveness/performance would be impacted since now nodes would need to fetch keys from other nodes to actually process blocks, and the development effort involved would be extremely high and would deprioritize other things.

This is a tough tradeoff between decentralization/liveness/performance vs the increased attack surface and potential privacy breaches.