In the previous post I addressed the drawbacks of FHE-based blockchains and why TEE can help improve it. In this post, I'd like to frame out what an FHE coprocessor looks like and how we can build an FHE coprocessor with TEE based on Phala's serverless wasm platform.

Definition of FHE coprocessor

A service that can help DApps do FHE computing, by generating keys and decrypting data with the key for DApps on behalf

Here is the site of Zama's key management service, the workflow of their coprocessor are:

- DApps submit a request to their standalone blockchain
- Off-chain coprocessor fetch request, do computing (forward to FHE computing server)
- coprocessor decrypts computing result, and submits to standalone blockchain
- DApps read results from blockchain
- Coprocessor is running inside Secure Enclave to guarantee the security of the whole network

More details can be found in their workshop at Ethcc. Here are a few things I'd like to highlight:

- A FHE coprocessor is not limited to serving FHE-based blockchain, but also DApps that have the requirements to do FHE computation
- An FHE coprocessor can be built on top of the existing blockchain, no need to build a new chain
- The purpose of submitting a request to a blockchain is to guarantee data integrity and verifiability, but we don't necessarily to submit a request to the blockchain, or other L1/L2, DA is also a good choice
- Technically there are two things the coprocessor will do after getting the returned encrypted data from the FHE server. First is decrypt the data with the proper key, and second is re-encrypt the real result with the user's account public key

, so that only a specific user can see the result (decrypt the result with their account key). Still, workflow can be different in different implementation

Implement A TEE FHE coprocessor

From the above explanations, we know that building an FHE coprocessor is actually building a FHE-friendly MPC network, things can be more complicated determined by how much friendly we want to be, see zama's paper. But generally, basic functions should be contained in FHE coprocessor are:

- Key generation and share between nodes (for example based on Shamir's secret sharing scheme)
- Decrypt data based on FHE decryption schema (Zama already provides WASM binding can be used in our JS engine)

People who are interested in the implementation details can check out our recently posted article.