Abstract.

*Privacy-preserving protocols such as Secret Network using TEEs gives granular control over contract metadata in the form of “Secret Contracts” which have encrypted input, state, and output. However, applications often have macro level data that needs to be available for users to track summary statistics important to the application (such as total liquidity, total tokens minted, etc.). This allows correlation to be drawn between a macro statistic changing and an individual interaction with a contract - ultimately revealing the underlying data without compromising the integrity of the encryption schema in any capacity. A proposed solution for this is epoch / periodic reveals of the state of a contract, making it difficult to draw any clear correlation between summary data and individual transactions.

The Problem

User interacts with a private contract by sending an encrypted input. The contract then updates the internal state of the contract. A public query that is a summary statistic is then queryable and reflects the change in the state of the contract. Any user tracking interactions with the contract are then able to form an easy correlation between the change in the publicly queryable data, and the user’s interaction with the contract. The net result can be complete disclosure of the contract interaction without ever compromising the encryption layer of the contract’s input or state or the users input. While public summary statistics are important to applications, this unfettered access to macro data ultimately defeats the purpose of encryption techniques by giving anyone scanning both contracts and summary data one to one correspondence.

[

1213×429 13 KB

](https://global.discourse-cdn.com/standard17/uploads/enigma1/original/2X/c/c14bd7daf777a931049b69988271ad62f196f86a.png)

Epoch Reveal

With an epoch reveal schema, the state of the contract that is publicly queryable is only updated within a certain time frame X. The trade-off for this is users have less of an accurate picture at any given moment of the underlying application summary data, but in return are granted an amount of privacy that is distinctly more difficult for any data tracking to uncover (assuming n>=2 number of contract interactions during the epoch obfuscation period).

[

1228×425 9.44 KB

](https://global.discourse-cdn.com/standard17/uploads/enigma1/original/2X/3/3a1c88d4b8480030067dccfe29e90e4d1dd931b3.png)

In the above example, three users interact with the contract, adding in {2,100,32} for a total macro statistic of 134 that is queryable at the end of the epoch privacy phase. Because the public contract state is only updated at the end of the epoch privacy phase at t = 1 hour, anyone tracking macro summary data will be unable to correlate how much any of the given users contributed to the contract, especially since all transactions are kept encrypted. The more users that interact with the contract between the epoch reveals of t=0 and t=1, the greater the anonymity set and therefore obfuscation from correlation. Additionally, the greater the length of time between t=0 and t=1, the more time there is for transactions to join the anonymity set. Because secret contracts have encryption by default, it takes a minimal amount of obfuscation and participation between epoch reveals to hide meaningful correlation. This is because there is: n! * x possible outcomes, where x is the number of possible permutations within the change in summary statistics and n is the number of entities interacting with the contract during the epoch privacy phase.

Conclusion

Combining the power of secret contracts working in parallel to epoch reveals of public data on private contracts brings a degree of unparalleled privacy to applications, while still maintaining control over summary statistics important for the end user experience. The use of SGX encryption combined with the simple epoch reveal schema outlined above could play a significant role in cross-chain bridge privacy, DeFi applications, and more. It would be proposed that such a technique should fundamentally be ingrained into a token specification and contract standard as an optional implementation for applications to use.

-Carter Woetzel