We propose to use DKG+DVT+RCO to let native staked ETH as RAI's CDP.

The major problem of using staked ETH as CDP is stakers can control their validators to make 51% attack by mint RAI before their attack to reduce their cost of attack. To create a secure solution, it is necessary to prevent stakers from controlling their own validator and also ensure the operators cannot launch attacks. Therefore, we propose combining DKG(Distributed Key Generation) with DVT(Distributed Validator Technology) and RCO(Random Chosen Operators) to build this platform and achieve RAI products based on ETH staking.

(This document aim to achieve the idea that Vitalik posted at RAI forum, I post it here hope to gather more community suggestions and hopefully we can work with the Reflexer team and the DVT team to solve this problem.

Can oracles double as co-stakers? How RAI-like systems might safely support staked ETH - General Discussion - Reflexer)

No LST!

Here is a brief description of several key technologies:

DKG

The DKG is built on top of verifiable secret sharing (VSS) but eliminates the need for a trusted party. Its correctness and privacy are guaranteed by homomorphic encryption. In this way, a signing key is generated for a validator jointly by the operators. But the signing keys are not known to anyone including all the operators even though they can make use of their respective KeyShares to participate in producing the signature for the validator.

Besides being used for signing, KeyShares are also used for Resharing. This mechanism allows for the redistribution of key shares to other operators in case certain operators go offline or for dynamic addition and removal of operators.

DVT

Distributed Validator Technology (DVT) is similar to a multi-signature consensus voting. It divides a Validator Key into multiple key shares and can aggregate signatures, allowing Ethereum PoS validators to operate on multiple nodes or machines.

It is important to select a mature DVT network for the underlying architecture. Currently, ssv.network appears to be a better choice as they provide a wide range of operators to choose from. However, their DKG technology is still under development. We have talked about this with their team, It is expected to be available shortly after the mainnet launch.

RCO(Random Chosen Operators)

To further ensure that the staking key won't be controlled by anyone, there should be a mechanism to randomly select operators for the DKG process. We can leverage an algorithm similar to Ethereum's Beacon Chain's random proposer selection algorithm to randomly choose multiple operators from a cluster of operators that meet certain criteria.

Option 1: Randomly select operators from the top fifty performers in terms of performance within the operator cluster of a DVT network.

Option 2: Determine the selection range based on the historical performance of operators, such as those with an effectiveness rating of 95% or above.

These options provide a random selection process while taking into account performance metrics or historical operator performance to ensure a fair and secure selection of operators for the DKG process.

Solution:

Develop a smart contract to support:

1. The staker transfer 32 ETH to this smart contract.

2. The smart contract calculates random numbers to randomly select operators.

2.1 For solo stakers who are willing to run their own staking nodes, the staker can run their own Ethereum clients and register two operators on ssv.network. During staking, they specify their two operators to be included among the four operators. The contract randomly selects the other two operators to match.

2.2 For stakers who do not wish to run their own Ethereum clients, the contract randomly selects four operators.

1. The selected operators execute DKG calculations to generate the staking key and key shares.

2. Call the deposit contract with staking key information to stake 32 ETH, with the withdrawal address set to this contract.

3. Call the RAI contract to generate a CDP and mint RAI.

4. If liquidation happened, trigger exit and transfer ETH to the liquidator.

Analysis for section 2.1 in solution:

Trust Analysis:

- Increased decentralization: Operators cannot fully control validators, eliminating the possibility of operator collusion.
- Operators cannot collude to individually slash stakers.
- Operators cannot collude to orchestrate proposer attacks because block proposals require staker signatures.
- Stakers cannot individually cause slashing or dominate attacks, thereby preventing CDP losses.
- Operators or stakers can go offline, in which case the contract triggers resharing or direct withdrawal for exiting.
- If both operators and stakers go offline, the contract can trigger resharing followed by immediate withdrawal.

It can be seen that this solution actually implements Vitalik’s Idea2, where the trust is shifted from trusting a single oracle to trusting two operators, and this trust is also limited.

Another flaw mentioned by Vitalik regarding Idea2 is the concern about intentional or unintentional offline behavior by oracles. However, in our solution, this flaw can be mitigated. Firstly, the top 50 operators are generally professional operators, unlike oracles that may be low-quality operators. Secondly, we can directly trigger resharing or withdrawal.

In the future, if 0x03 withdrawal Credentials can be deployed on the mainnet, triggering withdrawal will be even simpler.

Attack Cost Analysis:

Assuming a staking rate of 50% for RAI, an attacker seeking to exploit the system for their malicious purposes would need to control at least half of the validators to achieve lower attack costs compared to directly attacking the Ethereum network without utilizing this system:

For example, if an attacker holds 320 ETH, participating in this system by staking the entire amount would yield RAI tokens worth 160 ETH. In order to equalize or reduce the attack costs compared to directly attacking Ethereum with 320 ETH, the attacker must control at least five or more out of ten validators (320/32) to participate in the attack.

In this scenario, the attacker would need to control two operators themselves, and additionally, at least one operator must be randomly selected. However, achieving a position within the top 50 validators is already a challenging task, requiring substantial costs to maintain nodes and reputation. If we disregard these costs, an attacker would need to control 25 out of the top 50 operators to lower attack costs.

Analysis for section 2.2 in solution:

Trust Analysis:

- Further increases the difficulty of an attack for stakers.
- Operators cannot slash stakers individually; multiple operators must collaborate.
- In the event that all operators are offline, contract-triggered resharing or direct withdrawal occurs.

Attack Cost Analysis:

Assuming a staking rate of 50% for RAI, an attacker seeking to exploit the system for their malicious purposes would need to control at least half of the validators to achieve lower attack costs compared to directly attacking the Ethereum network without utilizing this system:

For example, if an attacker holds 320 ETH, participating in this system by staking the entire amount would yield RAI tokens worth 160 ETH. In this case, the attacker must control five or more out of ten validators (320/32) to participate in the attack. This would equalize or reduce the attack costs compared to directly attacking Ethereum with 320 ETH.

So, how many operators would an attacker need to control to manipulate half of the validators?

Let’s assume the total number of operators, n, and each time, m operators are randomly selected from n to operate a validator. The attacker controls k operators. The probability of the attacker fully controlling a validator is given by:

$P = (n-m)! * k! / n! / (k-m)!$

Assuming $n = 50$ and $m = 4$, calculations show that when $k = 42$, $P = 0.486$, and when $k = 43$, $P = 0.536$.

In other words, the attacker must control 43 out of the top 50 operators to lower attack costs.

Furthermore, we expect these operators to be teams that uphold Ethereum's consensus, and their reputation makes the probability of any of them violating Ethereum's consensus extremely low. Thus, the attack costs are significantly higher.

Regarding Vitalik's statement, "RAI holders already trust the oracle not to screw them over in this way, and if we trust oracles in other lower-stakes ways, it seems like that wouldn't add much vulnerability to the system," My understanding is that we believe in orcle because we believe in the professionalism and decentralization of it, for example, Chainlink is a decentralized oracle machine, but we can't believe that Chainlink can be a good stake operator just because we believe that they can do a good job of oracle, they may screw up because they have no stake experience.

I'm a researcher from PlanckerDAO, PlanckerDAO is a group of Engineers, Reseachers, PMs

from Asia who long for building the Ethereum eco

in the Great China, we'd like to cooperate with RAI and DVT team to solve this problem to make RAI better because we highly appreciate the philosophy of RAI, the crypto world needs a stablecoin that is completely decoupled from traditional finance.

Thanks for your time to read this.