TL;DR

Following the initial concept on https://governance.aave.com/t/arfc-aave-v3-caps-update-framework/11937/4, propose to the community a path forward on the idea of Risk Steward

s: smart contracts for risk providers to change parameters with strict on-chain validations defined in advance.

Risk Stewards: a progressive approach

Aave v3 enables the addition of new features concerning permissions granularity, due to its system of roles: RISK_ADMIN

, POOL_ADMIN

, EMERGENCY_ADMIN

, etc.

A Risk Steward is a smart contract that receives one or more of the aforementioned roles from the Aave governance, and by acting as an extra layer of indirection/validation, allows to introduce extra logic to control strictly what the “owner” of the Steward can do.

Technically, it is relatively simple for the community to give granular control over the management of all different parameters to Risk Stewards, but the complexity comes in terms of decentralization and self-protection of the protocol, even from entities that should be trustable, like service providers.

Therefore, we propose to start with the simplest Risk Steward case, but that will already simplify community voting overhead and operations: increase of supply and borrow caps

.

CapsPlusRiskSteward

The initial Risk Steward we propose will work the following way:

- A CapsPlusRiskSteward

smart contract will be created per each Aave v3 instance, ownable.

- The owner will be a 2-of-2 multisig: 1 signer being a representative of Gauntlet @Pauljlei), and another from @ChaosLabs.
- The owner will be able to call functions with permissions to increase supply & borrow caps of all assets in Aave v3 pools.
- The functions to increase caps will have the following limitations, enforced on-chain:
- The cap to increase should be strictly higher than the current one.
- For each asset, only 1 increase is allowed every 5 days.
- The supply cap can never be increased above 50% of the current one.
- The borrow cap can never be increased above 50% of the current one.
- The cap to increase should be strictly higher than the current one.
- For each asset, only 1 increase is allowed every 5 days.
- The supply cap can never be increased above 50% of the current one.
- The borrow cap can never be increased above 50% of the current one.
- No timelock on changes, as this is focused on operational optimization, and an increase in caps should not hurt the protocol.

From a design perspective, the objective of this initial iteration is not to have a perfect system, but a workable pilot with no downside, on which the community can iterate depending on the needs.

The procedure around the usage of this Steward is something that we leave for the risk providers to decide, but we think it should be compatible with the current practices and transparency, like informing the community in advance before doing changes.

FAQ

2-of-2 multisig? What if one of the members loses access to the signer wallet?

- Not a problem. The Aave governance is still the only entity that can grant and remove roles from other addresses. In that kind of situation, a new Steward controlled by a new 2-of-2 can be deployed, permissions removed from the previous, and granted to the new one.

How much trust will be given to a Steward and the entity in control of it?

- This is a trust-minimized system. The actions the entity controlling it can execute on the Aave protocol are exclusively those allowed by the logic of the Steward smart contract itself.

What if the entity controlling the Steward decides to upgrade its logic?

- This first Steward (and highly probably applicable to all future ones) will be IMMUTABLE, not possible to upgrade its logic.

The entity controlling it can’t do anything apart from what is defined in the smart contract.

Why not include caps decrease too?

- Caps decrease could represent in practice a freezing of a pool. It is something to consider in the following stages, but we consider it a way more complex decision compared with the increasing scenario.

Why not include more entities of the community?

- In this model, trust minimization is managed in the implementation phase of the Steward and on the governance vote to give to it the required roles.

The community has explicitly selected 2 risk providers, so adding extra parties in for example a 2-of-3 multisig would give quite important role to the extra actor in deciding the outcome of some changes.

But sometimes Gauntlet and Chaos Labs have disagreed on recommendations. Will the community get locked when that happens?

- It is perfectly healthy for risk contributors to disagree, as they have different underlying models and tools. Whenever that happens (not so frequently with caps increases), the current mechanism of on-chain governance proposal can be applied, only sacrificing speed.

Next steps

As always, we would like to receive any feedback from the community, apart from what was commented in the initial post.

Afterward, targeting the end of this week, we will create a Snapshot vote, and if the outcome is positive, will be followed by the implementation of the smart contract and the on-chain proposal that will grant RISK_ADMIN

to it, in all the networks.