Racks Protocol

0xSoda, 0xKinando, nathan.eth

Abstract:
Racks Protocol is an omnichain issuer of stablecoins and liquid staked derivatives collateralized by a permissionless yield aggregation engine. Unlike other stablecoin issuers, Racks constantly rotates the exposure of collateral to different yield venues, thus maximizing the returns available to the user. Unlike other aggregators, Racks allocates capital to multiple strategies per asset simultaneously - reducing yield dilution as user deposits scale - while also operating in an entirely chain-agnostic manner. Racks will initially issue RXD and rxETH, stablecoins pegged to USD and ETH respectively.

Keywords:
Yield Aggregation — Asset Management — Omnichain Liquidity — Stablecoins — Liquid Staked Derivatives

Contents

Introduction

Stablecoins are defined as cryptocurrencies whose value is pegged, or tied, to that of another currency, commodity, or financial instrument. They maintain peg through active purchasing and redeeming for the underlying assets, most often dollars for centralized stablecoins or Ethereum for their decentralized counterparts. Recently, stablecoin issuers have begun investing their collateral into low-risk investments in order to provide a native yield for their stablecoins. Yield Aggregators are defined as trustless fund management platforms designed to rotate capital between an approved set of yield generating venues, based on whichever one provides greatest yield per asset.

Racks is a DeFi savings account and issuer of stablecoins, powered by the first universal interface for cross-chain yield aggregation, structured derivative products, and custom automated strategies. It is designed to make sophisticated, multi-step yield generation strategies simple and composable; enabling DeFi users of any experience to earn industry-leading yield without depositing into a CeFi black box. Unlike protocol's other vaults, deposits to the Racks Ether and Racks Dollars Vaults return rxETH and RXD respectively; liquid stablecoins pegged to ETH and USD.

Motivation

Dollar-denominated stablecoins, Ether, and their derivatives are often the highest volume assets across cryptocurrency exchanges, lending markets, and payment providers. Accordingly, there exists a larger opportunity cost on the depositing of these assets into any platform, as there is often another opportunity for yield that can fluctuate higher. Racks aims to minimize opportunity cost on Ether and Dollar deposits, by issuing a liquid stablecoin directly to the user's wallet. All rewards earned on the underlying deposits are used to fully collateralize the minting and distributing of new tokens to users — making earning yield just as passive as holding dollars or ether.

Racks does not issue stablecoins as Collateralized Debt Positions - there is no obligation to repay the stablecoins to the vault, no inefficiently low LTV to mint tokens against. Minting RXD and/or rxETH is most akin to a receipt token used on Aave - passively accruing yield in the form of new tokens in the user wallets. Unlike receipt tokens, however, Racks tokens are liquid and just as useful as their underlying assets.

1. Protocol Design Overview

1.1 Asset Management

Assets deposited into the Racks Protocol are managed entirely by the Racks Machine — an entirely on-chain asset management system operated by a set of smart contracts on each blockchain Racks operates upon, responsible for distributing inflow capital between different strategies while minimizing dilution of yield per strategy. The Racks Machine adjusts target weights per strategy based on the Racks Controller — an off-chain data analytics machine, it ingests on-chain data relating to strategy performance and liquidity in order to output target weights per strategy. Safeguards exist to prevent arbitrary access or manipulation of funds within the Racks Protocol, and all capital management operations are executed entirely by smart contracts.. In time, the protocol intends to bring the Controller entirely on-chain as well, eliminating any perceived risk vectors for user capital.

1.2 Natively Omnichain Tokens

rxETH and RXD differ from traditional stablecoins in the sense that their collateral is often decoupled from the chains on which they are liquid. Technically, they are more similar to liquid derivatives of Ether and Dollar denominated vaults than stablecoins, yet function identically due to the liquidity available for them. The Racks Dollar and Racks Ether vaults accept deposits from any chain, and enable users to receive RXD and rxETH on any chain. Therefore, RXD and rxETH function as a chain-agnostic source of liquidity for any assets - enabling rapid bootstrapping of new blockchain ecosystems through instant and slippage-free asset mobility, without compromising on exposure to better yields found elsewhere.

1.3 Peg Stability Mechanisms

rxETH and RXD maintain peg across multiple liquidity pools on multiple blockchains through a combination of organic market stabilization and automated arbitrage. Organic market stabilization occurs principally through demand for the tokens at a discount, conviction in the availability of collateral, and continued demand for the tokens at the pegged price. Market Stabilization is the preferred method of maintaining the peg; it has no impact on TVL, and creates additional opportunities to earn return on investment for market participants. Automated arbitrage begins to occur when price diminishes beyond the Market Stabilization Range — the amount the token may depeg to the downside before arbitrage begins. Arbitrage within the protocol occurs by using the capital collateralizing the stablecoins, which is always greater than the price of the stablecoin in the event of depegging, in order to repurchase and burn RXD/rxETH. This process continues until the stablecoin in question has returned to peg, while all arbitrage profits captured by the purchase of tokens at a discount increase protocol revenues.

2. Protocol Architecture

2.1 Racks Machine

The Racks Machine serves as a cross chain execution and asset management system, consisting of sub machines deployed on each network. The machine also manages issuance of Racks vault tokens, and peg stability. The Central Machine operates from Ethereum Mainnet - serving as an accounting hub. Identical Sub-Machines are deployed across secondary chains.

The machine directs user capital across multiple programmed strategies, based on weights programmed by the off-chain Racks Controller - which works to minimize yield dilution of new deposits.

When new weights are tabled by the Racks Controller, The Central Machine organizes cross chain reweights via the Reweight Sequencer by :

- Emitting new target weights across the entire system
- Conducting lowest slippage operations in the reweight immediately using pre-programmed strategy functions
- Conducting larger risk mitigating operations immediately

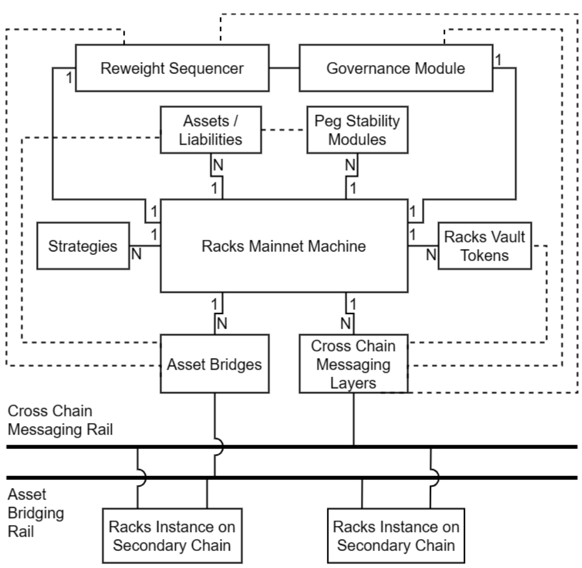

Figure 1: Diagram showing relationship between different components of the mainnet racks instance. Connections to cross chain rails are also shown. Secondary chain instances are similar to the mainnet, however, are designed to receive cross chain commands from mainnet.

2.2 Cross-Chain Rails

Instances of the Racks Machine need to exchange both assets and data cross-chain. None of the solutions on the market provide full coverage across the ecosystem, thus it is imperative to build in support for as many solutions as possible. In cases where a given needed chain pair is supported by multiple bridges or messaging layers - Racks benefits from more competitive rates and redundancy. The asset bridge modules will use the cheapest bridge available at all times, and cross-chain messaging modules will adopt redundancy patterns whereby messages are sent simultaneously via multiple channels. Building redundancy into cross-chain operations means less vulnerability to a given messaging layer being exploited - embodying decentralization in the architecture of the protocol itself. Racks is modular and upgradeable enough to integrate new messaging layers at any point, giving it security beyond that of any protocol using singular messaging layers, single-point data sources, or single external keeper functions to manage protocol affairs.

2.3 Automated Arbitrage Defense

When RXD or rxETH diminish from their parity prices by more than 3%, collateral from within the vaults is automatically deployed to repurchase and burn tokens. If collateral is c, then only $0.97c$ is needed to repurchase each RXD or rxETH - leaving some surplus collateral after price parity has been restored and supply has been deflated. This overcollateralization acts as a future insurance fund against depegging or smart contract risk.

3. Strategy Weighting Calculations

Using market data to weight strategies is key to maximizing yield whilst managing risk. These calculations will initially occur in an off-chain environment, the Racks Controller, however, will be fully transparent.

3.1 Ingesting Strategy Data

In order to incorporate a strategy into a given vault's investment mix, we must gather the following data via analysis of on-chain activities.

- Principal asset (e.g. USDC)
- Smart Contract Risk for both principal asset and strategy
- Depeg risk of principal asset
- TVL/Yield curve given current demand factors
- Current TVL

For the sake of simplicity, we will ignore external factors that drive yield, such as demand for borrowing assets or changes in reward token value.

3.2 Risk Probability

The total risk probability of a given strategy A over a given time period, can be represented as

$P_A = P(E_{A\_S} \mid E_{A\_P} \mid D_A)$, where :

- P_A = total risk probability
- E_{A_S} = strategy exploit
- E_{A_P} = principal token exploit
- D_A = complete depeg of principal token (to zero)

It can be naively assumed, for the sake of risk management, that any one of these events occurring could result in a complete loss of funds.

3.3 Yield vs Total Strategy Deposits

By analyzing factors affecting strategy yield, such as utilization ratios + interest rate models and reward token emissions - we can create a curve $Y_A(d)$ which returns the expected yield given a total strategy deposit of d. This curve is the basis of all capital rebalancing functions, as it determines how much of total deposits per asset are allocated to each strategy per asset.

3.4 Determining Market Risk appetite

By analyzing the levels of deposits across all strategies provided within the wider ecosystem, we can determine the net risk appetite of the market. We can do this by calculating :

$\frac{\sum_{r=0}^{n} P_r d_r}{\sum_{r=0}^{n} d_r}$ where r is the index of a given strategy, and d is the amount of capital deposited into it.

By modeling this data over time, we can determine the long term average risk level of the market and use this to account for the movement of capital by external players following our reweight calculations. Moving forward we shall refer to this long term risk variable as P_{MARKET}.

3.5 Optimal Capital Distribution

Given P_{MARKET} , we can estimate the ideal distribution of capital amongst strategies by finding the highest yielding permutation that is closest to the target risk profile. We can score a given permutation, p, such that :

$S_P = \log([\mathrm{abs}(P_p - P_{MARKET})^{-1}]) * Y_p$

Doing so, we determine the ideal permutation to represent logical market behavior[1], which will serve as an input variable for determining how to best allocate capital for Racks. Many permutations would score highly, however, we will select one that requires realistic levels of capital movement.

This set can be represented as $\{d_{M0}, d_{M1}, \ldots d_{Mn}\}$.

3.6 Assembling Strategy Weightings

We use the following processed data to assemble our strategy weights :

- Y(d) curves for each given strategy
- P for each given strategy
- The optimum market capital distribution
- Rack's own risk appetite (P_{RACKS})

Given Racks' desire to position itself as higher yield, but somewhat higher risk, in comparison to traditional financial offerings, $P_{RACKS} > P_{MARKET}$. We will find a high yielding permutation of Racks capital distribution given the optimum market capital distribution set, such that :

$\sum_{r=0}^{n} Y(d_{mr} + d_{Rr})$ is maximized while keeping net risk below P_{RACKS}.

Cost of reweighting comes into play as a practical factor which could impede returns, especially long term as many reweights will occur - thus reweights moving less capital are preferred. As such, we will consider this tradeoff when choosing the optimal permutation, or whether to wait for opportunity costs to grow to justify said reweight.

[1] Although cryptocurrency markets can hardly be considered to follow "logical market behavior", we believe that as the industry grows, it will trend towards becoming a more efficient market.