

# Ithil

*A generalised leveraged investment strategies protocol*

V1.0.0 - February 4, 2022

## **Abstract**

Ithil aims at introducing undercollateralised leveraged strategies in DeFi - a game changer for traders, liquidity providers and other protocols who can now rely on a variety of investment products to address their needs. Modular and upgradable at its core, Ithil offers users and other protocols leveraged interactions with the DeFi space, enabling an entirely new range of trading opportunities, acting as an open box financial instrument open to everyone. Simplicity is the key quality of Ithil, allowing everybody from the DeFi novice to sophisticated experts to access many investment strategies.

# 1 Introduction

The DeFi has recently seen an incredible growth and expansion in the user base, seeing the first institutional players come in too. Built on the premises of being an open space where composability and cross-contamination of protocols sits at its core, access to DeFi is getting more difficult as protocols grow in complexity and require extensive research from their users to come up with truly earning strategies. It is commonly seen how LPs on yield aggregators like Beefy Finance need to often move their liquidity to the newest strategies when the previous ones reach a breakdown point where the APY dramatically falls to 0. Similarly, uneducated users who just want a *quick and dirty* solution may find themselves losing their capital even when using supposedly "safe" blue-chip protocols like Uniswap.

## 1.1 What is Ithil

At its core, Ithil is built around the concept of under-collateralised lending, allowing a leverage greater than 1. Leveraged investments are a concept that has been around for centuries, allowing traders to invest more than their available liquidity. Leverage makes it possible for users to take advantage of profitable market opportunities even if they don't have immediate access to the funds required. However, it comes with the risk of the trader losing multiples more than in the case of a classical, non-leveraged investment. As always, there is no free lunch.

Ithil enables leveraged interactions with other DeFi protocols in a composable way, offering to its users a curated set of customisable investment strategies to choose from. For example, margin trading using an external dex such as Uniswap or Kyber, or leveraged staking on Yearn, are possible on Ithil using the resident liquidity as principal. In every strategy, the trader pays a *collateral* to be able to use an amount of the protocol's liquidity higher than the collateral itself to open a *position*; such collateral is refunded together with the earnings obtained when such position is closed.

In case a leveraged position is incurring a loss which would cause the protocol's liquidity to be at risk, Ithil uses a decentralised network of bots to liquidate it and prevent impaired loans from becoming detrimental to the protocol itself. Liquidators are compensated when a losing position is closed by them.

Compared to other standard lending protocols [CITAZIONE], the under-collateralization of loans forces borrowed funds and tokens obtained as a result of the investment to never leave the protocol. When a token is exchanged for another one through an external protocol, such as a dex or a staking one, the obtained tokens are transferred to the strategy contract: only at the closure of the position, the profit and loss is computed and the consequent transfers are performed.

## 1.2 Why is Ithil different

There are many existing protocols [CITAZIONE] trying to go beyond traditional overcollateralised lending but fail to offer a seamless user experience and remain limited to the realm of a few highly-educated and wealthy DeFi users [CITAZIONE]. Moreover, unlike most protocols [CITAZIONE] which only use

stablecoins-backed loans, Ithil uses *value-agnostic vaults*. This means that the ultimate goal of a vault with a given ERC20 token, is to increase its balance in *that token*; therefore, the vault's dollar value is an immaterial parameter. In this way, Ithil can provide loans backed by any ERC20 token, gaining in this way also the capacity of being a profitable staking protocols for tokens which are difficult to stake elsewhere [CITAZIONE/ESEMPIO]. The precise list of tokens supported by Ithil is community-based, and its rational is preventing malicious tokens to be injected into the protocol.

## 2 The Vault

As mentioned, leverage is made possible by using the protocol's liquidity. The contract in which liquidity is stored and managed is the *vault*: liquidity providers (LPs) can stake their assets in the vault to provide the necessary liquidity, which will be then used by investors, who will pay fees that will be distributed back to LPs. At the eyes of LPs, Ithil is therefore a staking protocol, in which any asset can be staked to generate APY. This is more flexible than other protocols [CITAZIONE] in which only stablecoins or a restricted class of assets can generate income.

On the technical side, the various investment strategies borrow funds from the vault through the `borrow` function, and repay the loan when the position is closed calling the `repay` function. Since these functions will move the vault's funds, they are not callable by anyone but the strategies themselves, whose addresses are stored in an array within the vault to implement whitelisting. Such array `strategies` can be modified only by governance vote, to allow a new strategy (or disallow an old one) to use the vault's liquidity.

When a strategy is called, the margin is transferred as a form of payment to the vault or to the strategy, depending on the particular strategy used. [DIAGRAMMA]. The strategy then borrows from the vault the amount of capital required to perform the desired strategy and calls the relevant external protocols. When a position is closed, the vault's loan and relative fees are repayed by the strategy. See the strategies dedicated section for more details.

Ithil's vault is *value-agnostic*: once an amount of XYZ tokens are staked, we do not track the relative value (in USD or an underlying peg) of such treasury; instead, we simply assure the generation of more XYZ tokens.

The vault is responsible for the following tasks:

- Handling staking and unstaking.
- Calculation of the interest rate applied when taking a loan.
- Lending tokens to one of Ithil's supported strategies.
- Keeping account of the net loans and insurance reserve balance.

The vault also contains a list of *whitelisted tokens* approved by the governance that can be provided by LPs or borrowed by the traders. When LPs stake liquidity into the vault, they receive `iTokens`, Ithil's interest-bearing tokens representing liquidity put to work in the vault and can be used on other protocols.

## 2.1 Fee Redistribution

Ithil generates income in the form of fees paid by the traders. Such fees are then redistributed to the LPs proportionately on how much liquidity they provided by accruing value in the vault and increasing the backing of the yield-bearing **iToken**: LPs will only get rewards in the specific token they provided [DIAGRAMMA]. When tokens are lent for a strategy, the vault calculates the fixed fees and the interest rate applied to the loan. The fixed fees are token-specific, community-based percentages of the amount which goes out of the vault, and are the direct compensation to LP for providing their liquidity to Ithil.

In all cases, fees are considered when deciding whether to liquidate a position, so that the payment is assured for any trader's profit or loss scenario. In particular, a *liquidation factor*  $r$  is set for every position, such that the position is liquidated if the trader's payback goes below  $r\%$  of the margin.

### 2.1.1 Ithil's LP tokens

When an investor deposits an amount  $D_{\text{TKN}}$  of ERC20 tokens TKN in Ithil's vault, an amount  $M_{\text{iTKN}}$  of LP tokens **iTKN** are minted and transferred to the investor. Letting  $S_{\text{iTKN}}$  be the total supply of **iTKN**,  $B_{\text{TKN}}$ ,  $L_{\text{TKN}}$  and  $I_{\text{TKN}}$  the vault's TKN balance, net loans, and insurance reserve balance respectively, then the amount of tokens to be minted is calculated as follows:

$$M_{\text{iTKN}} = \frac{S_{\text{iTKN}}}{B_{\text{TKN}} + L_{\text{TKN}} - I_{\text{TKN}}} D_{\text{TKN}} \quad (1)$$

Similarly, an investor who owns a quantity  $M_{\text{iTKN}}$  of LP tokens **iTKN**, can redeem them any time to get  $D_{\text{TKN}}$  TKN, still calculated using Equation (1); in this case, the vault will burn the investor's tokens. In the particular case in which the total balance of the vault is zero (vault's initialization for the particular token), then we mint wrapped tokens in a 1 : 1 ratio.

The quantity

$$\sigma_{\text{TKN}} := \frac{D_{\text{TKN}}}{M_{\text{iTKN}}}$$

is a function of the current vault's state and total LP token supply (and not on the particular deposit/withdraw amount), and we call it the *share price* for token TKN. A higher share price means that more tokens can be redeemed by burning LP tokens, so we can say that the share price captures the amount of fees generated by the platform.

### 2.1.2 Absence of arbitrage

Notice that this system is arbitrage-free: assume (we omit the indices not to make the notation too heavy) another investor deposits  $D'$  after the first one, thus getting  $M'$  wrapped tokens according to Equation (1), and that no fees have been generated in the meantime. This will increase the vault's balance by  $D'$  and the total wrapped token supply by  $M'$ . According to Equation (1), the first investor can now redeem

$$D_1 = \frac{B + D' + L - I}{S + M'} M = \frac{B + D' + L - I}{S + \frac{S}{B+L-I} D'} M = \frac{B + L - I}{S} M \quad (2)$$

which is exactly the amount of tokens the first investor could redeem before the second one deposited, i.e.  $D_1 = D$ . Therefore, deposits or withdrawals from other investors do not affect the share price, and in particular, if an investors deposits and withdraws immediately some funds, he or she will get exactly the same amount that has been deposited. This means that the fees generated before one's own deposit cannot be claimed, and arbitrage is impossible.

## 2.2 Interest Rate

The interest rate is investment-specific and represents the risk beyond the investment as seen from the lender's perspective. Both the interest rate and the fixed fee applied are returned as a parameter from the dedicate vault's functions, while the repayment and the logic for the calculation of risk is included in the implementation of the investment strategies. The riskiness is captured in an integer called *risk factor*  $\beta$  (see following section) which is passed to the vault's `borrow` function as a parameter. The interest rate is then computed as

$$r = \left(1 + \frac{\max(L - I; 0)}{B}\right) \frac{(r_{\text{base}} + \beta)}{\kappa} \quad (3)$$

Here  $r_{\text{base}}$  is the basic commonly-shared interest rate applied to all tokens and decided by the governance,  $\kappa$  is the *collateralization* of the loan (i.e. the ratio between the collateral posted by the trader, and the amount borrowed from the vault),  $B$  is the vault's balance,  $L$  is the net loan amount, and  $I$  is the insurance reserve balance. Notice that if the position is entirely covered by its margin, i.e.  $\kappa \rightarrow \infty$ , then the interest rate becomes null (but the vault's fixed fee is always applied). This is consistent with the fact that no value has been borrowed from the vault. Notice also that the fees increase with the net loans and decrease with the insurance reserve balance: when too many loans have been granted, it will become expensive to open a new position, thus decreasing the value at risk of the vault.

## 3 Strategies

We can think of strategies as configurable actions executed across several other protocols in a composable way. They span from basic swaps to convoluted lending and farming of liquidity pools' tokens. In general, the community can develop a strategy and whitelist it into Ithil; conversely, strategies can be delisted from Ithil by a community decision.

In order to be able to open a position in any strategy, a trader has to post a collateral to cover for potential losses caused by unfavorable market movements. The collateral can be posted in any token, and potentially gives the right to use any token type of the ones within the vault. The maximum loan size is determined by imposing that the interest rate, as of Equation (3), should not exceed a maximum interest rate  $r_{\text{max}}$  fixed by the governance; this is the same of saying that

$$\kappa \geq \left(1 + \frac{\max(L - I; 0)}{B}\right) \frac{(r_{\text{base}} + \beta)}{r_{\text{max}}}$$

For example, in the scenario of a very risky investment (high risk factor), or when most of the vault is uncovered (insurance pool amount too low), the allowed leverage is lower.

### 3.1 Margin trading

The first strategy we consider and support is Margin Trading. This is a classical investment strategy in which a trader borrows some tokens in an undercollateralized way, and swaps them on a dex to speculate on a price change of a particular token pair. When such swap is performed, a *position* is opened, which is stored onchain on Ithil's contract. The amount of tokens borrowed is called the *principal* of the position, while the amount obtained from the swap is the *allowance*.

Since the loan is undercollateralized, the allowance does not actually goes into the trader's hand. It is more convenient to say that the trader has bought the "right" to move the balance from the dealer's account to and from an external dex (which we will consider fixed).

We define the *ratio*  $r$  as the exchange value of TKB/TKA as published by the dex. This can be seen as the number of tokens TKA one can obtain by exchanging one unit of TKB. Notice that things in reality are much more complicated, since the very action of exchanging tokens in the dex will move this ratio; we will not discuss these subtleties now and simply assume that if we exchange  $x$  TKB on the dex, we will get  $xr$  TKA. Ithil has quoters implemented to get precise estimates of an amount obtained by the dex, which are used to perform liquidation calculations (see later).

#### 3.1.1 Long position

Assume the trader believes that the value of the ratio  $r = 10$  is about to *increase* in the future, and that his belief is strong enough that he is willing to post a 100TKA worth of margin and going into a *long position* on TKB with a leverage of 10. At this point, the vault checks if there are 1000TKA available. If it is the case, it will exchange them to buy 100 TKB from the dex. We assume a fee of 1% on the vault relative to TKA, and a risk factor of 50% for liquidation. The vault then registers 900TKA in the net loans: they must be repayed when the position is closed.

**Scenario of increasing  $r$**  Assume the day after we observe  $r = 11$ , i.e. an increase of 10% with respect to the previous day. At this point, the trader can choose to close his position: the vault will sell back the 100TKB to the dex, but now 1100TKA will be obtained for this amount. The debt of 900TKA is repayed and the vault will refund the trader with the extra 200TKA, minus 10TKA representing 1% of the swapped amount, which represent the fee of the vault. Therefore the trader will go home with 190TKA, that is a gain of 90% of the original investment when the market has only shown a 10% increase. This happens because of the leverage of 10 the trader has chosen to undertake. To summarize:

- Trader's initial balance: 100TKA, vault's initial balance: 900TKA.
- Trader's final balance: 190TKA, vault's final balance: 910TKA.

**Scenario of decreasing  $r$ : no liquidation** Assume the day after we observe  $r = 9.8$ , i.e. a decrease of 2% with respect to the previous day. At this point, the trader can choose to close his position: the vault will sell back the 100TKB to the dex, but now only 980TKA will be obtained for this amount. Similarly to what happened before, the vault's debt of 900TKA is repayed and the remaining amount of 80TKA, minus 10TKA of fees, are transferred to the trader, which will go home with 70TKA, realising a loss of 30% of the original investment.

- Trader's initial balance: 100TKA, vault's initial balance: 900TKA.
- Trader's final balance: 70TKA, vault's final balance: 910TKA.

**Scenario of decreasing  $r$ : liquidation** Assume the day after we observe  $r = 9.5$ , i.e. a decrease of 5% with respect to the previous day. At this point, we notice that selling back the 100TKB to the dex, one can only obtain 950TKA. Since the margin is of 100TKA, the fees are 10TKA, and the risk factor has been set to 50%, the position will be *liquidated*, i.e. closed by a liquidator, who then repays 900TKA plus fees to the vault, and gets the remaining margin of 40TKA.

- Trader's initial balance: 100TKA, vault's initial balance: 900TKA.
- Trader's final balance: 0TKA, vault's final balance: 910TKA.

### 3.1.2 Short position

Assume the trader believes that the value of the ratio  $r = 10$  is about to *decrease* in the future, and that his belief is strong enough that he is willing to post a 100TKA worth of margin and going into a *short position* on TKB with a leverage of 10. At this point, the vault with native token TKB checks if it has 1000TKA worth of TKB in its pool as of today's market condition: i.e. the vault checks if it has 100TKB. If it does, it exchanges them to obtain 1000TKA from the dex and registers 100TKB net loans. Since the trader already posted 100TKA of margin, the allowance for this position will be 1100TKA. We assume a fee of 1% on the vault relative to TKA, and a risk factor of 50% for liquidation.

**Scenario of decreasing  $r$**  Assume the day after we observe  $r = 9$ , i.e. a decrease of 10% with respect to the previous day. At this point, the trader can choose to close his position selling back a quantity of TKA necessary to repay the debt 100TKB, plus 1TKB representing the fee. As of today, only 909TKA are necessary. The extra 191TKA will go to the trader, that is a gain of 91% of the original investment when the market has only shown a 10% decrease.

- Trader's initial balance: 100TKA, vault's initial balance: 100TKB.
- Trader's final balance: 191TKA, vault's final balance: 101TKB.

(Notice that the trader has 1TKA more than the analogous scenario we discussed for a long position: this is caused by the fact that the margin posted is in a token, which is different from the one borrowed from the vault. Since the fees are always in the vault's borrowed token, the trader will profit or lose also on the fees for the exchange ratio movements).

**Scenario of increasing  $r$ : no liquidation** Assume the day after we observe  $r = 10.2$ , i.e. an increase of 2% with respect to the previous day. At this point, the trader can choose to close his position: the vault will sell back a

quantity of TKA necessary to obtain back 101TKB to restore the liquidity and get the fees, but now 1030.2TKA are necessary. The trader will then obtain 69.8TKA by closing this position:

- Trader's initial balance: 100TKA, vault's initial balance: 100TKB.
- Trader's final balance: 69.8TKA, vault's final balance: 101TKB.

**Scenario of increasing  $r$ : liquidation** Assume the day after we observe  $r = 10.5$ , i.e. an increase of 5% with respect to the previous day. At this point, the vault notices that 1060.5TKA are now necessary to obtain 101TKB to have back the liquidity plus fees. Since the margin is of 100TKA, and the risk factor is set to 50%, the position will be *liquidated*. As in the long case, a liquidator will close the position and grab the remaining 39.5TKA.

- Trader's initial balance: 100TKA, vault's initial balance: 100TKB.
- Trader's final balance: 0TKA, vault's final balance: 101TKB.

### 3.2 Other possibilities

Similar systems allow long trading with non-native token margin, and short trading with native token margin. In any case, the vault's gain will be of 1%, while the trader will be more exposed to the exchange ratio if the margin posted is of a non-native type.

## 4 Cross-chain support

Ithil will support provisioning liquidity from multiple evm-compatible chains as well as running strategies on other chains too. For instance, LPs can provide liquidity from Ethereum mainnet, Polygon, Avalanche, Harmony, BSC and strategies can be run on all those L2 and sidechains thanks to Composable finance *Personas*.

## 5 Governance

In a first initial phase governance will be shared among the core team using a multiple key and a multisig contract. The admins will be able to harvest fees, vary the fee ratio and correct the risk factor but cannot alter in any way the current open positions nor interfere with the liquidity pools. Since contracts are immutable and their implementation cannot be changed, a governance token may be needed to push new functionalities and vote on key parts of the protocol itself, like the choice of the dex. The release of a governance token is left to a second phase in the protocol life when a critical mass of traders, LPs and community DAO members is reached.

Ithil promotes a thriving developer community, encouraging the creation of trading bots as well as the interaction with third party smart contracts to manage investments within the limits of the code itself.

In a further step, the aforementioned administrative powers of this contract will be further limited by putting the protocol into a "owner-less" mode, where the core team relinquish control over admin functions to a voting system, thus



making Ithil a DAO. A lack of centralized power is essential to the trustlessness of the protocol.

## **6 Risks**

## **7 Competitors**

## **8 Acknowledgments**

We would like to thank the incredible Ethereum community for its support and welcoming atmosphere as well as ETHGlobal for running hackathons and onboarding new developers while creating connections with key projects in the DeFi space like Uniswap or Yearn finance.