SOLIDITY.*FINANCE*

VenturesDAO - Smart Contract Audit Report

SUMMARY

VenturesDAO is building a platform to maximize yields across DeFi protocols for investors.

For this audit, we reviewed VenturesDAO's contracts at commit 5359201778e6d97d2be61ee7756a2baad714a4f3 on GitHub. The current codebase includes a user-facing Vault contract, and a strategy contract for integration with Yearn Finance.

The contracts are deployed at the following addresses on the Ethereum

- Yearn-Farmer USDT v2 - 0x3DB93e95c9881BC7D9f2C845ce12e97130Ebf5f2
- Yearn-Farmer USDC v2 - 0x4F0C1c9bA6B9CCd0BEd6166e86b672ac8EE621F7
- Yearn-Farmer DAI v2 - 0x4A9dE4dA5eC67E1dbc8e18F26E178B40D690A11D
- Yearn-Farmer TUSD v2 - 0x9f0230FbDC0379E5FefAcca89bE03A42Fec5fb6E

- DAO Vault Medium USDT - 0x3685fB7CA1C555Cb5BD5A246422ee1f2c53DdB71
- DAO Vault Medium USDC - 0x2bFc2Da293C911e5FfeC4D2A2946A599Bc4Ae770
- DAO Vault Medium DAI - 0xA6F1409a259B21a84c8346ED1B0826D656959a54
- DAO Vault Medium TUSD - 0x2C8de02aD4312069355B94Fb936EFE6CFE0C8FF6

> *Notes on the Contracts:*
>
> - *Users can deposit stablecoins into the vault contracts to earn interest on their assets.*
> - *These deposited funds will be migrated by the team into a strategy contract to earn interest and rewards. The "Yearn*

Yearn Earn contract or to a Yearn Vault contract; both earning rewards from the Yearn Finance platform.

- *The contract will mint daoUSDT/daoDAI/etc., representing the user's stablecoins staked in the contract.*
- *The DAOVentures ecosystem has a seperate contract for each stablecoin (UDST, UDSC, TUSD, DAI); though the logic of them are the same.*

- *The Yearn Finance contracts optimises the interest accrual process across dYdX, AAVE, and Compound to obtain the highest interest rate for users.*
- *The Yearn contracts will issue the Yearn Farmer contract yUSDT which chall accrue interest over time. This accrued interest will ultimatley flow back to depositers when they withdraw.*
- *The team can set the strategy address at any time except for a two day window prior to migrations.*
- *The team can call the vesting() function on the strategy contract, which will withdraw all of the deposits from the Earn and Vault contracts, combined with the interest rewrads earned, and hold them in the Farmer strategy contract.*
- *Once vesting() is called, users will be able to claim their deposited tokens and accrued interest via the Vault contract.*
- *Upon withdrawing, platform/development fees will be deducted from the amount and the remaining balance transferred to the*

- *Fees incurred depend on the amount deposited. The team will set a lower and upper bound, thereby creating three fee tiers - below the bottom bound, in between the two, and above the upper bound. The team sets the associated fee rates for each range.*
- *The project team can update the platform fees at any time, though they had imposed a limit on these values to prevent abuse. Vault addresses can also only be set once to prevent abuse.*

- *The 'before' and 'after' logic in the refund() function in the vault can be removed if the strategy refund() function provides a return value of the shares to be burned.*
- *The approvePooling() function's if statments can be safely removed to save gas on calls and deployment costs.*
- *The 'shares' local variable in the deposit() function is defined as 0, and then later added to. To save gas on each deposit, the shares local variable should be defined at the first point of addition, rather than initialized at the beginning of the call.*
- *Some functions could have been declared external instead of public to save on gas.*
- *Utilization of SafeMath throughout the platform to prevent overflow issues.*

> - *No security issues from external attackers were identified.*
> - *Ensure trust in the project team as they have notable power in the ecosystem.*
> - *Date: April 2nd, 2021*
> - *Update Date: April 7th, 2021 - Deployment to mainnet.*

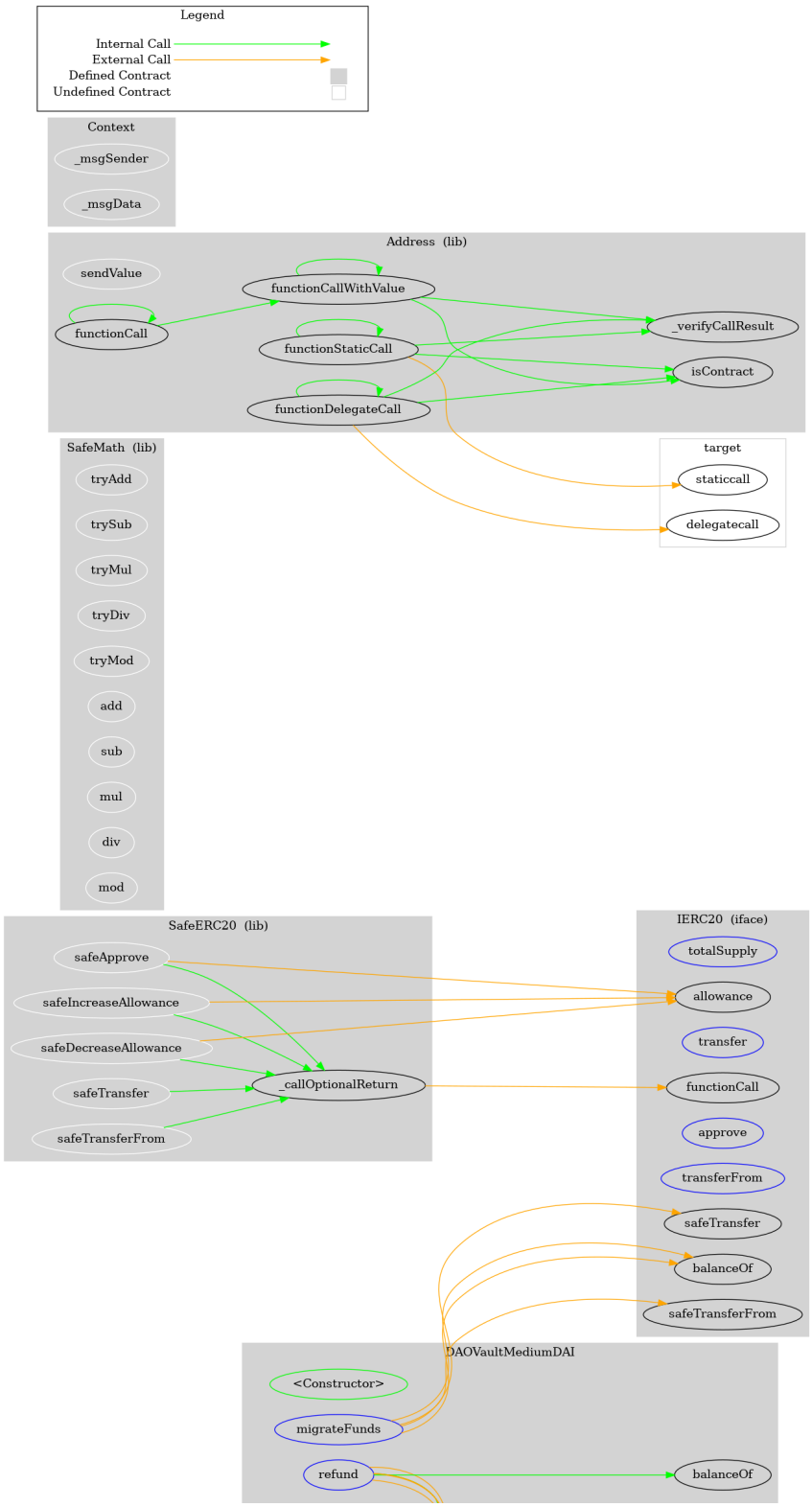
COMBINED EXTERNAL THREAT RESULTS

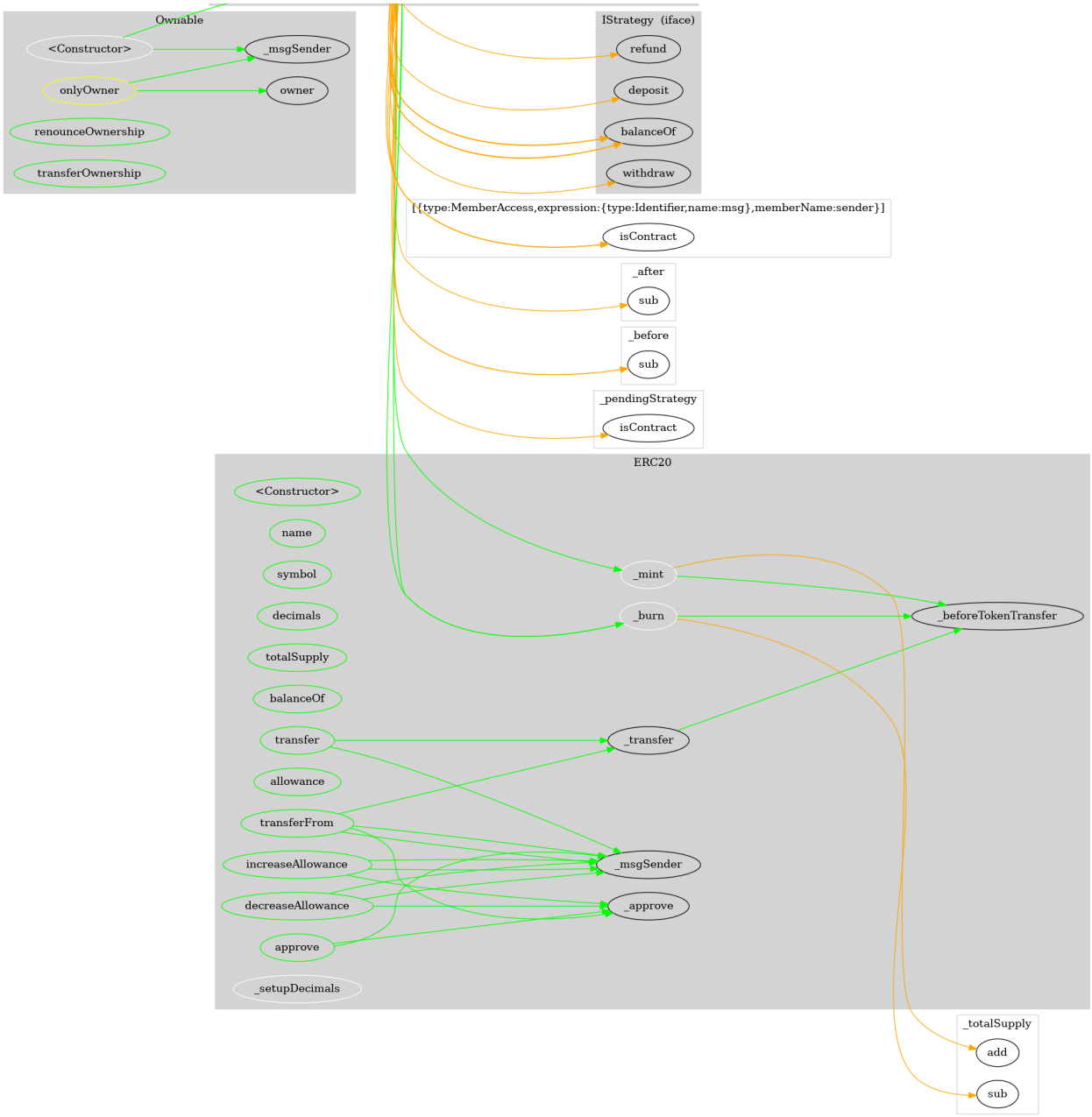
Vulnerability Category	Notes	Result
Arbitrary Storage Write	N/A	PASS
Arbitrary Jump	N/A	PASS
Delegate Call to Untrusted Contract	N/A	PASS
Dependence on Predictable Variables	N/A	PASS
Deprecated Opcodes	N/A	PASS
Ether Thief	N/A	PASS

Vulnerability Category	Notes	Result
External Calls	N/A	PASS
Integer Over/Underflow	N/A	PASS
Multiple Sends	N/A	PASS
Suicide	N/A	PASS
State Change External Calls	N/A	PASS
Unchecked Retval	N/A	PASS
User Supplied Assertion	N/A	PASS
Critical Solidity Compiler	N/A	PASS
Overall Contract Safety		PASS

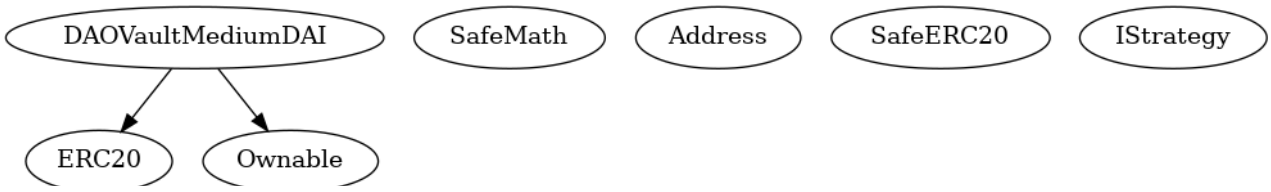
Details: DAO Vault Medium (DAI example)

FUNCTION GRAPH





INHERITENCE CHART



FUNCTIONS OVERVIEW

```
($) = payable function
# = non-constant function

Int = Internal
Ext = External
Pub = Public

+ [Int] IERC20
   - [Ext] totalSupply
   - [Ext] balanceOf
   - [Ext] transfer #
   - [Ext] allowance
   - [Ext] approve #
   - [Ext] transferFrom #

+ [Lib] SafeMath
   - [Int] tryAdd
   - [Int] trySub
   - [Int] tryMul
   - [Int] tryDiv
   - [Int] tryMod
   - [Int] add
   - [Int] sub
   - [Int] mul
```

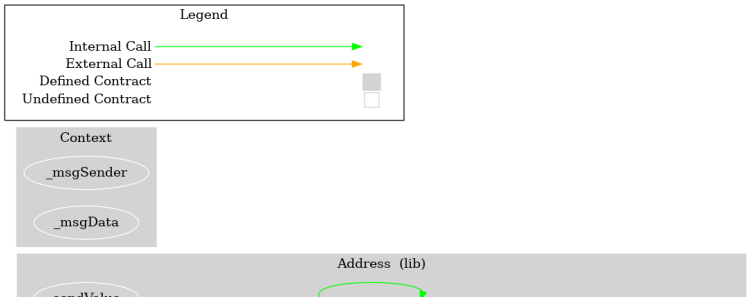
```
    - [Int] div
    - [Int] mod

+ [Lib] Address
    - [Int] isContract
    - [Int] sendValue #
    - [Int] functionCall #
    - [Int] functionCall #
    - [Int] functionCallWithValue #
    - [Int] functionCallWithValue #
    - [Int] functionStaticCall
    - [Int] functionStaticCall
    - [Int] functionDelegateCall #
    - [Int] functionDelegateCall #
    - [Prv] _verifyCallResult

+ [Lib] SafeERC20
    - [Int] safeTransfer #
    - [Int] safeTransferFrom #
    - [Int] safeApprove #
    - [Int] safeIncreaseAllowance #
    - [Int] safeDecreaseAllowance #
    - [Prv] _callOptionalReturn #

+  Context
    - [Int] _msgSender
    - [Int] _msgData

+  ERC20 (Context, IERC20)
```

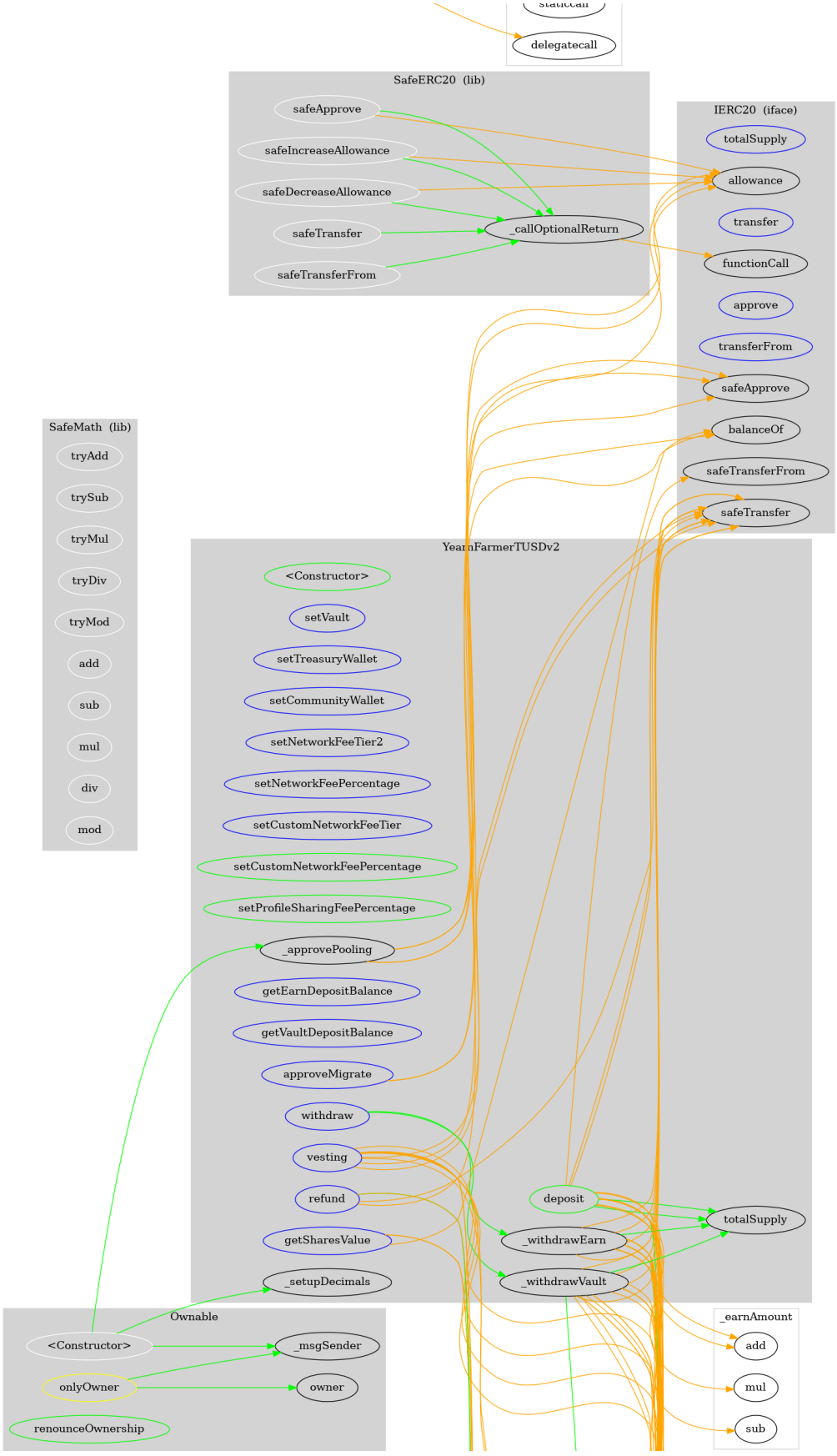
```
    - [Pub] decimals
    - [Pub] totalSupply
    - [Pub] balanceOf
    - [Pub] transfer #
    - [Pub] allowance
    - [Pub] approve #
    - [Pub] transferFrom #
    - [Pub] increaseAllowance #
    - [Pub] decreaseAllowance #
    - [Int] _transfer #
    - [Int] _mint #
    - [Int] _burn #
    - [Int] _approve #
    - [Int] _setupDecimals #
    - [Int] _beforeTokenTransfer #

+  Ownable (Context)
    - [Int]  #
    - [Pub] owner
    - [Pub] renounceOwnership #
       - modifiers: onlyOwner
    - [Pub] transferOwnership #
       - modifiers: onlyOwner

+ [Int] IStrategy
    - [Ext] deposit #
    - [Ext] withdraw #
    - [Ext] refund #
    - [Ext] balanceOf
```

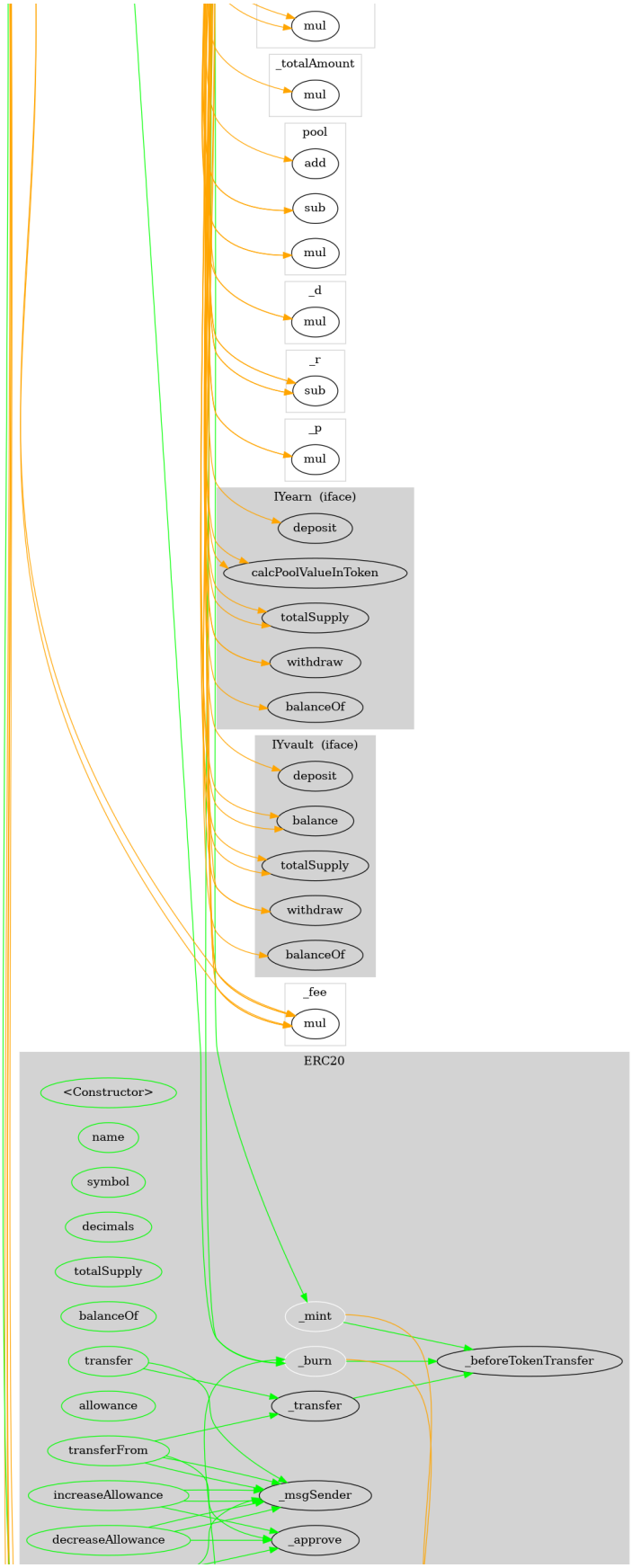
```
      - modifiers: ERC20
  - [Ext] deposit #
  - [Ext] withdraw #
  - [Ext] refund #
  - [Ext] setPendingStrategy #
      - modifiers: onlyOwner
  - [Ext] unlockMigrateFunds #
      - modifiers: onlyOwner
  - [Ext] migrateFunds #
      - modifiers: onlyOwner
```

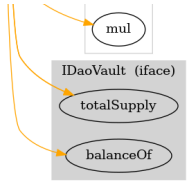
Details: Yearn Farmer Strategy (TUSD example)

FUNCTION GRAPH

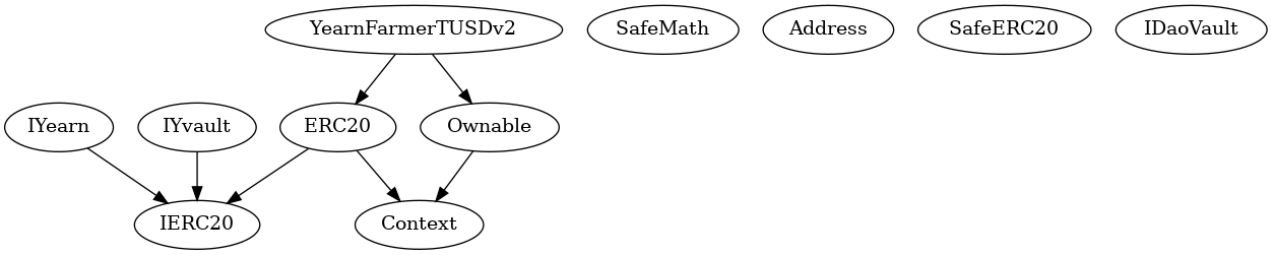








INHERITENCE CHART



FUNCTIONS OVERVIEW

```
($) = payable function
# = non-constant function

Int = Internal
Ext = External
Pub = Public

+ [Int] IERC20
   - [Ext] totalSupply
   - [Ext] balanceOf
```

```
    - [Ext] transferFrom #

+ [Lib] SafeMath
    - [Int] tryAdd
    - [Int] trySub
    - [Int] tryMul
    - [Int] tryDiv
    - [Int] tryMod
    - [Int] add
    - [Int] sub
    - [Int] mul
    - [Int] div
    - [Int] mod
    - [Int] sub
    - [Int] div
    - [Int] mod

+ [Lib] Address
    - [Int] isContract
    - [Int] sendValue #
    - [Int] functionCall #
    - [Int] functionCall #
    - [Int] functionCallWithValue #
    - [Int] functionCallWithValue #
    - [Int] functionStaticCall
    - [Int] functionStaticCall
    - [Int] functionDelegateCall #
    - [Int] functionDelegateCall #
    - [Prv] _verifyCallResult
```

```
    - [Int] safeTransferFrom #
    - [Int] safeApprove #
    - [Int] safeIncreaseAllowance #
    - [Int] safeDecreaseAllowance #
    - [Prv] _callOptionalReturn #

+   Context
    - [Int] _msgSender
    - [Int] _msgData

+   ERC20 (Context, IERC20)
    - [Pub]  #
    - [Pub] name
    - [Pub] symbol
    - [Pub] decimals
    - [Pub] totalSupply
    - [Pub] balanceOf
    - [Pub] transfer #
    - [Pub] allowance
    - [Pub] approve #
    - [Pub] transferFrom #
    - [Pub] increaseAllowance #
    - [Pub] decreaseAllowance #
    - [Int] _transfer #
    - [Int] _mint #
    - [Int] _burn #
    - [Int] _approve #
    - [Int] _setupDecimals #
    - [Int] _beforeTokenTransfer #
```

```
    - [Pub] owner
    - [Pub] renounceOwnership #
       - modifiers: onlyOwner
    - [Pub] transferOwnership #
       - modifiers: onlyOwner

+ [Int] IYearn (IERC20)
    - [Ext] calcPoolValueInToken
    - [Ext] deposit #
    - [Ext] withdraw #

+ [Int] IYvault (IERC20)
    - [Ext] balance
    - [Ext] deposit #
    - [Ext] withdraw #

+ [Int] IDaoVault
    - [Ext] totalSupply
    - [Ext] balanceOf

+  YearnFarmerTUSDv2 (ERC20, Ownable)
    - [Pub]  #
       - modifiers: ERC20
    - [Ext] setVault #
       - modifiers: onlyOwner
    - [Ext] setTreasuryWallet #
       - modifiers: onlyOwner
    - [Ext] setCommunityWallet #
       - modifiers: onlyOwner
```

```
    - modifiers: onlyOwner
- [Ext] setCustomNetworkFeeTier #
    - modifiers: onlyOwner
- [Pub] setCustomNetworkFeePercentage #
    - modifiers: onlyOwner
- [Pub] setProfileSharingFeePercentage #
    - modifiers: onlyOwner
- [Prv] _approvePooling #
- [Ext] getEarnDepositBalance
- [Ext] getVaultDepositBalance
- [Pub] deposit #
- [Ext] withdraw #
- [Prv] _withdrawEarn #
- [Prv] _withdrawVault #
- [Ext] vesting #
    - modifiers: onlyOwner
- [Ext] getSharesValue
- [Ext] refund #
- [Ext] approveMigrate #
    - modifiers: onlyOwner
```

GO HOME

Please review our Terms & Conditions, Privacy Policy, and other legal information here. By using this site, you explicitly agree to these terms.