Definitex Token

Code Security Assessment

PREPARED BY:

THE AUDIT INSTITUTE ANALYST TEAM

PREPARED FOR:

THE DEFINITEX TEAM

PREPARED ON:

MARCH 20TH 2021

THE AUDIT INSTITUTE

Report Version 1.0

Table of Contents

Disclaimer

The Audit Institute Reports are not, nor should be considered, an “endorsement” or “disapproval” of any particular project or team. These reports are not, nor should be considered, an indication of the economics or value of any “product” or “asset” created by any team or project that contracts The Audit Institute to perform a security review.

The Audit Institute Reports do not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technology’s proprietors, business, business model or legal compliance.

The Audit Institute Reports should not be used in any way to make decisions around investment or involvement with any particular project. These reports in no way provide investment advice, nor should be leveraged as investment advice of any sort.

The Audit Institute Reports represent an extensive auditing process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology.

Blockchain technology and cryptographic assets present a high level of ongoing risk. The Audit Institute's position is that each company and individual are responsible for their own due diligence and continuous security. The Audit Institute's goal is to help reduce the attack vectors and the high level of variance associated with utilizing new and consistently changing technologies, and in no way claims any guarantee of security or functionality of the technology we agree to analyze. View our full legal terms and conditions at https://audit.institute/

What is included in a report by *The Audit Institute*?

- A document describing the detailed analysis of a particular piece(s) of source code provided to The Audit Institute by a Client.
- An organized collection of testing results, analysis and inferences made about the structure, implementation and overall best practices of a particular piece of source code.
- Representation that a Client of The Audit Institute has indeed completed a round of auditing with the intention to increase the quality of the company/product's IT infrastructure and or source code.

Overview

Project Summary

Project Name & Website	**Definitex** - **Https://definitex.org**
Project Description	The Audit Institute team reviewed the token contract of the Definitex platform. The goal of the token is to provide investors with a financial vehicle to fuel their staking platform. This is designed to encourage users to gain passive value while holding the token. The total token supply is 350,000 DFX.
Platform	Ethereum, Solidity
Compiler Version	^0.6.6
Mainnet Address	0xf1f5De69C9C8D9BE8a7B01773Cc1166D4Ec6Ede2
Delivery Date	March 20th 2021
Method of Audit	Static Analysis, Fuzzing, Manual Review
Consultants Engaged	2

Summary of Findings

Critical	0
Medium	0
Informational	1
Total Issues	**1**

Executive Summary

This Audit Report exclusively covers the analysis that was conducted on Definitex's token contract written in Solidity. The Audit Institute analysts completed a separate review of the Definitex Staking contract where the findings varied in criticality as some were related to Solidity code standards and optimization, while others put users at risk of losing their funds. As a result, the Definitex team is currently working on revisions to address those vulnerabilities.

The Definitex Token (DFX) is an ERC20 token that uses 18 decimals and currently has a total supply of 350,000 tokens. At the time of writing this report, 40.8% of that supply is in the Uniswap v2 LP.

Disclosed in the report below is a full analytical review of the Definitex Token contract after undergoing various test scenarios and code review. Our findings based on the token contract alone were limited to an optimization recommendation.

Contracts in Scope

CONTRACT NAME	CONTRACT DESCRIPTION
basicToken.sol	The Definitex Token Contract

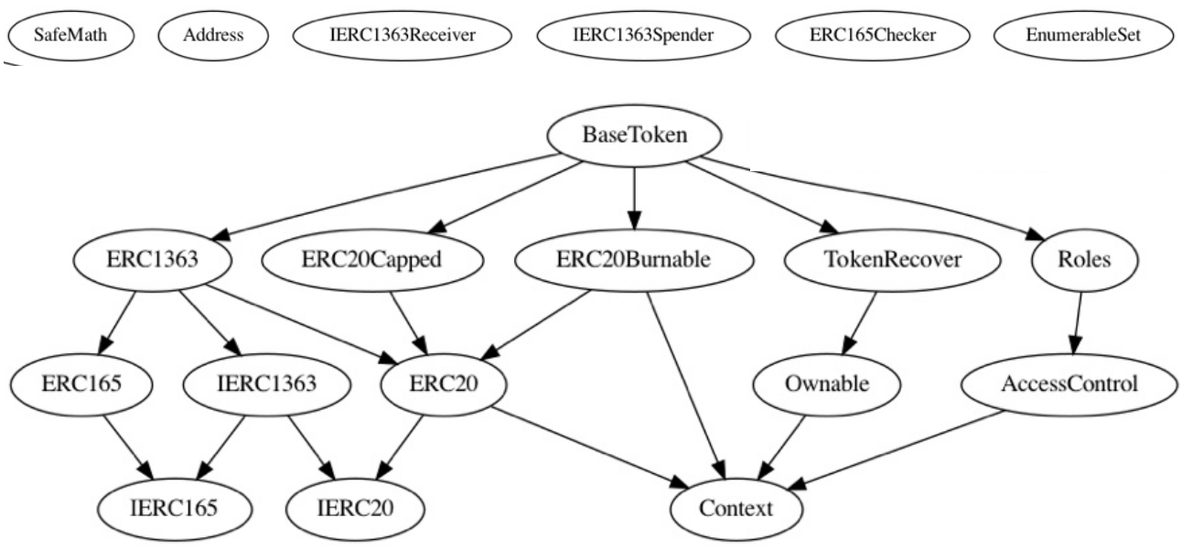
External Vulnerability Findings

Vulnerability Category	Notes	Results
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 / Token Loss	N/A	PASS
Exceptions	N/A	PASS
External Calls	N/A	PASS
External Service Providers	N/A	PASS
Flash Loans	N/A	PASS
Inconsistent Emission of Events	N/A	PASS
Integer Over/Underflow	N/A	PASS
Multiple Sends	N/A	PASS
Oracles	N/A	PASS
Reentrancy Issues	N/A	PASS
Unchecked Retval	N/A	PASS
Suicide	N/A	PASS
State Change External Calls	N/A	PASS
Unchecked Retval	N/A	PASS

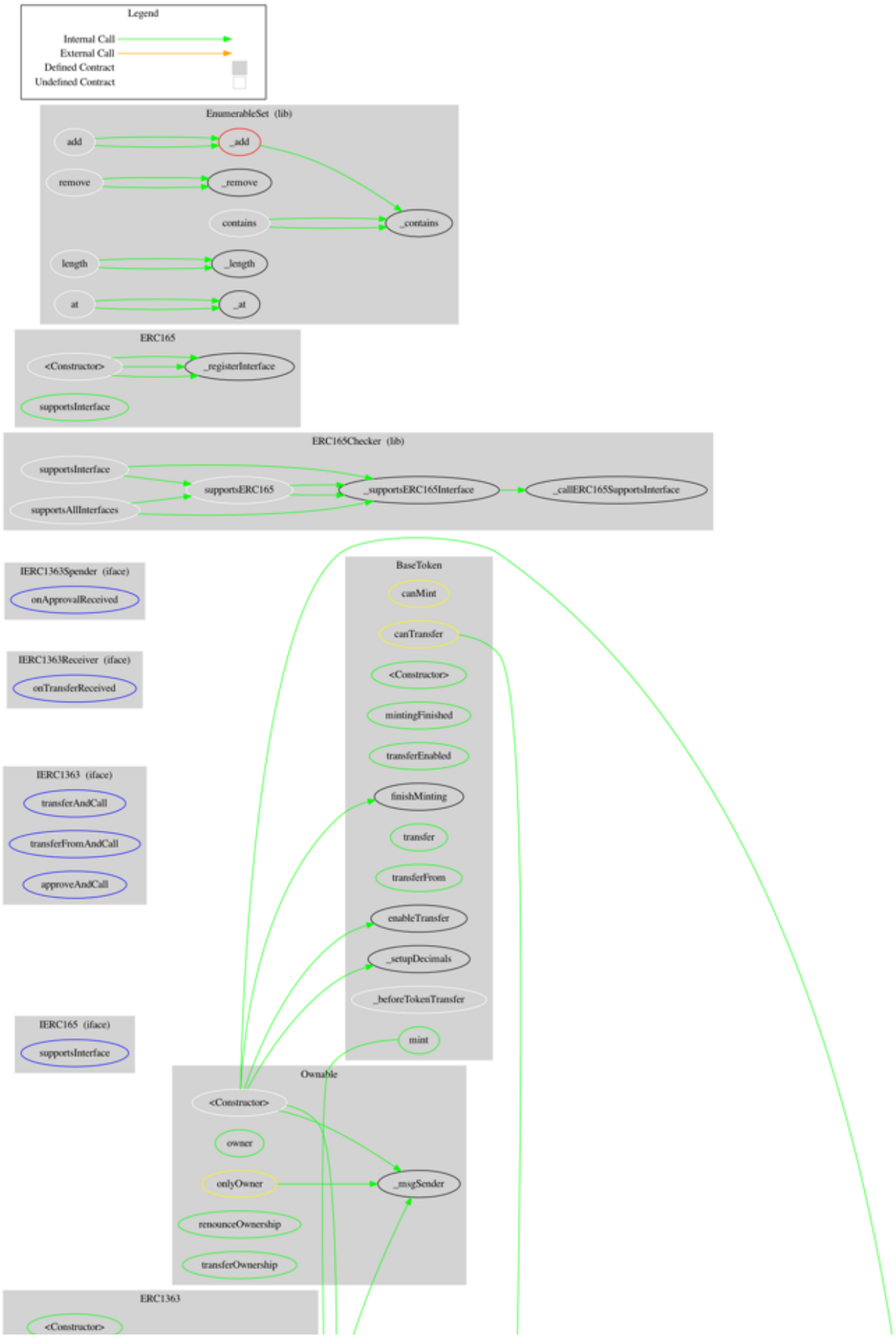
Findings and Recommendations

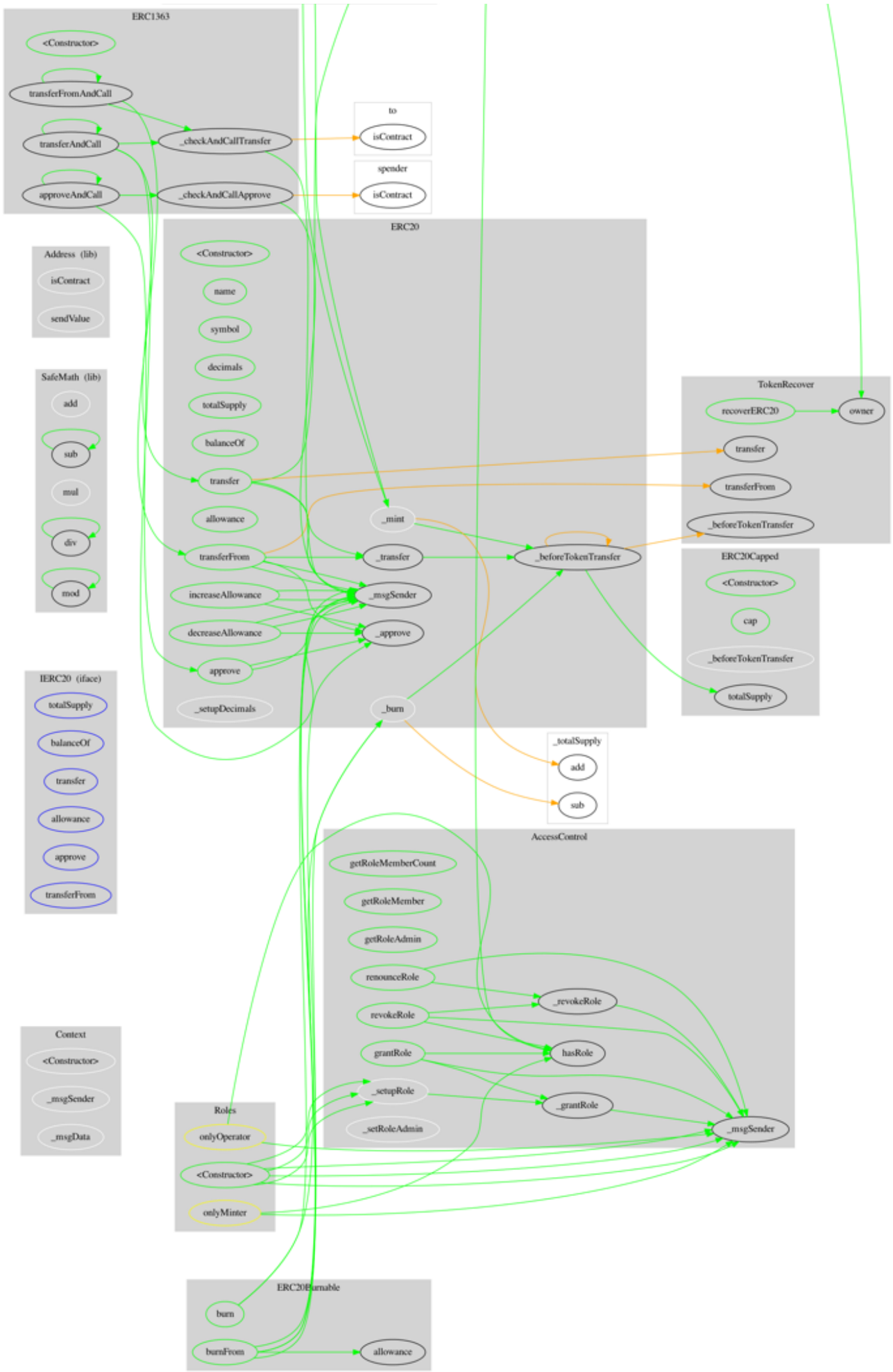
Finding Name	Criticality	Analyst Notes
Functions should be `external` (Gas Optimization)	Informational	BaseToken.mintingFinished() (Line# 1889-1891) BaseToken.transferEnabled() (Line# 1896-1898) BaseToken.mint(address,uint256) (Line# 1905-1907) **Recommendation: set these functions to external to slightly reduce gas cost.*

Inheritance Graph



Control Flow




ERC1363
<Constructor>
transferFromAndCall
transferAndCall
approveAndCall
_checkAndCallTransfer
_checkAndCallApprove
to
isContract
spender
isContract
ERC20
<Constructor>
name
symbol
decimals
totalSupply
balanceOf
transfer
allowance
transferFrom
increaseAllowance
decreaseAllowance
approve
_setupDecimals
_mint
_transfer
_msgSender
_approve
_burn
_beforeTokenTransfer
Address (lib)
isContract
sendValue
SafeMath (lib)
add
sub
mul
div
mod
IERC20 (iface)
totalSupply
balanceOf
transfer
allowance
approve
transferFrom
TokenRecover
recoverERC20
owner
transfer
transferFrom
_beforeTokenTransfer
ERC20Capped
<Constructor>
cap
_beforeTokenTransfer
totalSupply
_totalSupply
add
sub
AccessControl
getRoleMemberCount
getRoleMember
getRoleAdmin
renounceRole
revokeRole
grantRole
_setupRole
_setRoleAdmin
_revokeRole
hasRole
_grantRole
_msgSender
Context
<Constructor>
_msgSender
_msgData
Roles
onlyOperator
<Constructor>
onlyMinter
ERC20Burnable
burn
burnFrom
allowance

Functions Overview

($) = payable function
 # = non-constant function
Int = Internal
Ext = External
Pub = Public

+ Context
- [Int] <Constructor> #
- [Int] _msgSender
- [Int] _msgData

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

+ [Lib] SafeMath
- [Int] add
- [Int] sub
- [Int] sub
- [Int] mul
- [Int] div
- [Int] div
- [Int] mod
- [Int] mod

+ [Lib] Address
- [Int] isContract
- [Int] sendValue #

+ ERC20 (Context, IERC20)
- [Pub] <Constructor> #
- [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 #

+ ERC20Capped (ERC20)
- [Pub] <Constructor> #
- [Pub] cap
- [Int] _beforeTokenTransfer #

+ Ownable (Context)

+ ERC20Burnable (Context, ERC20)
- [Pub] burn #
- [Pub] burnFrom #

+ [Int] IERC165
- [Ext] supportsInterface

+ [Int] IERC1363 (IERC20, IERC165)
- [Ext] transferAndCall #
- [Ext] transferAndCall #
- [Ext] transferFromAndCall #
- [Ext] transferFromAndCall #
- [Ext] approveAndCall #
- [Ext] approveAndCall #

+ [Int] IERC1363Receiver
- [Ext] onTransferReceived #

+ [Int] IERC1363Spender
- [Ext] onApprovalReceived #

+ [Lib] ERC165Checker
- [Int] supportsERC165
- [Int] supportsInterface
- [Int] supportsAllInterfaces
- [Prv] _supportsERC165Interface
- [Prv] _callERC165SupportsInterface

+ ERC165 (IERC165)
- [Int] <Constructor> #
- [Pub] supportsInterface
- [Int] _registerInterface #

+ ERC1363 (ERC20, IERC1363, ERC165)
- [Pub] <Constructor> ($)
  - modifiers: ERC20
- [Pub] transferAndCall #
- [Pub] transferAndCall #
- [Pub] transferFromAndCall #
- [Pub] transferFromAndCall #
- [Pub] approveAndCall #
- [Pub] approveAndCall #
- [Int] _checkAndCallTransfer #
- [Int] _checkAndCallApprove #

+ TokenRecover (Ownable)
- [Pub] recoverERC20 #
  - modifiers: onlyOwner

+ AccessControl (Context)

- [Int] <Constructor> #
- [Pub] owner
- [Pub] renounceOwnership #
  - modifiers: onlyOwner
- [Pub] transferOwnership #
  - modifiers: onlyOwner

+ [Lib] EnumerableSet
- [Prv] _add #
- [Prv] _remove #
- [Prv] _contains
- [Prv] _length
- [Prv] _at
- [Int] add #
- [Int] remove #
- [Int] contains
- [Int] length
- [Int] at

- [Pub] hasRole
- [Pub] getRoleMemberCount
- [Pub] getRoleMember
- [Pub] getRoleAdmin
- [Pub] grantRole #
- [Pub] revokeRole #
- [Pub] renounceRole #
- [Int] _setupRole #
- [Int] _setRoleAdmin #
- [Prv] _grantRole #
- [Prv] _revokeRole #

+ Roles (AccessControl)
- [Pub] <Constructor> #

+ BaseToken (ERC20Capped, ERC20Burnable, ERC1363, Roles, TokenRecover)
- [Pub] <Constructor> #
  - modifiers: ERC20Capped,ERC1363
- [Pub] mintingFinished
- [Pub] transferEnabled
- [Pub] mint #
  - modifiers: canMint,onlyMinter
- [Pub] transfer #
  - modifiers: canTransfer
- [Pub] transferFrom #
  - modifiers: canTransfer
- [Pub] finishMinting #
  - modifiers: canMint,onlyOwner
- [Pub] enableTransfer #
  - modifiers: onlyOwner
- [Int] _beforeTokenTransfer #

END OF REPORT