

SKELETON ECOSYSTEM

SMART CONTRACT AUDIT



CATDOG ETH
ERC20

0xd13f785057B221b124dd02c459Aa09549EfD095e



Table of Contents

Table of Contents	1
Disclaimer	2
Overview	3
Creation/Audit Date	3
Verified Socials	3
Contract Functions Analysis	4
Contract Safety and Weakness	6
Detected Vulnerabilities	8
Contract Flow Chart	13
Inheritance Graph	14
Contract Descriptions	15

Global Disclaimer

This document serves as a disclaimer for the crypto smart contract audit conducted by Skeleton Ecosystem. The purpose of the audit was to review the codebase of the smart contracts for potential vulnerabilities and issues. It is important to note the following:

Limited Scope: The audit is based on the code and information available up to the audit completion date. It does not cover external factors, system interactions, or changes made after the audit. The audit itself can not guarantee 100% safety and can not detect common scam methods like farming and developer sell-out.

No Guarantee of Security: While we have taken reasonable steps to identify vulnerabilities, it is impossible to guarantee the complete absence of security risks or issues. The audit report provides an assessment of the contract's security as of the audit date.

Continued Development: Smart contracts and blockchain technology are evolving fields. Updates, forks, or changes to the contract post-audit may introduce new risks that were not present during the audit.

Third-party Code: If the smart contract relies on third-party libraries or code, those components were not thoroughly audited unless explicitly stated. Security of these dependencies is the responsibility of their respective developers.

Non-Exhaustive Testing: The audit involved automated analysis, manual review, and testing under controlled conditions. It is possible that certain vulnerabilities or issues may not have been identified.

Risk Evaluation: The audit report includes a risk assessment for identified vulnerabilities. It is recommended that the development team carefully reviews and addresses these risks to mitigate potential exploits.

Not Financial Advice: This audit report is not intended as financial or investment advice. Decisions regarding the use, deployment, or investment in the smart contract should be made based on a comprehensive assessment of the associated risks.

By accessing and using this audit report, you acknowledge and agree to the limitations outlined above. Skeleton Ecosystem and its auditors shall not be held liable for any direct or indirect damages resulting from the use of the audit report or the smart contract itself.

Please consult with legal, technical, and financial professionals before making any decisions related to the smart contract.

Overview

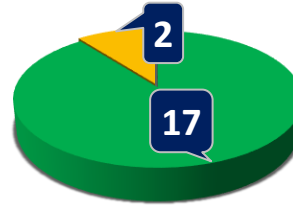
Contract Name	CATDOG
Ticker/Symbol	CADO
Blockchain	Ethereum ERC20
Contract Address	0xd13f785057B221b124dd02c459Aa09549EfD095e
Creator Address	0x919B5Cda44b889787e99F290059b7c09A4ACdC0d
Current Owner Address	Renounced
Contract Explorer	https://etherscan.io/token/0xd13f785057B221b124dd02c459Aa09549EfD095e
Compiler Version	v0.8.17+commit.8df45f5f
License	MIT License
Optimisation	Yes with 1500 rUNS
Total Supply	1,000,000,000 CADO
Decimals	18

Creation/Audit

Contract Deployed	05-Aug-2023
Audit Created	19-Aug-23 15:00:00 UTC
Audit Update	V 0.1

Verified Socials

Website	https://catdogtoken.net/
Telegram	https://t.me/catdogcommunity
X	https://twitter.com/catdog_coin



Contract Function Analysis



Pass



Attention Item















Risky Item

















Pass

Attention

Risk

Contract Verified		The contract source code is uploaded to blockchain explorer and is open source, so everybody can read it.
Contract Ownership		The ownership of the contract was sent to dead address. With this the owner eliminates he's rights to modify the contract. The owner can not set any of the functions anymore.
Buy Tax	3,1%	Shows the taxes for purchase transactions. Above 10% may be considered a high tax rate. More than 50% tax rate means may not be tradable. Contract renounced so tax rate is fixed. 
Sell Tax	3,1%	Shows the taxes for sell transactions. Above 10% may be considered a high tax rate. More than 50% tax rate means may not be tradable. Contract renounced so tax rate is fixed. 
Honeypot Analyse		Holder is able to buy and sell. If honeypot: The contract blocks sell transfer from holder wallet. Multiple events may cause honeypot. Trading disabled, extremely high tax
Liquidity Status		Locked on 18.08.2023: 98% for 84days Note! Initial liquidity tokens scanned. For new LP Lockers allways re-check with skeleton scanner on telegram.
Trading Disable Functions		No trading suspendable function found. If a suspendable code is included, the token maybe neither be bought or sold (honeypot risk). If contract is renounced this function can't be used.  If there is authorised hidden owner, or there is Retrieve Ownership Function, the trading disable function may be used!
Set Fees function		No Fee Setting function found. The contract owner may contain the authority to modify the transaction tax. If the transaction tax is increased to more than 49%, the tokens may not be able to be traded (honeypot risk). If contract is renounced this function can't be used.  If there is authorised hidden owner, or there is Retrieve Ownership Function, the set fees function may be used!
Proxy Contract		The proxy contract means contract owner can modify the function of the token and possibly effect the price. The Owner is not the creator but the creator may have authorisation to change functions.
Mint Function		No mint function found. Mint function is transparent or non-existent. Hidden mint functions may increase the amount of tokens in circulation and effect the price of the token. Owner can mint new tokens and sell. If contract is renounced this function can't be used.

Balance Modifier Function		<p>No Balance Modifier function found.</p> <p>If there is a function for this, the contract owner can have the authority to modify the balance of tokens at other addresses. For example revoke the bought tokens from the holders wallet. Common form of scam: You buy the token, but it's disappearing from your wallet.</p> <p> If contract is renounced this function still can be used as auto self Destruct</p>
Whitelist Function		<p>No Whitelist Function Found.</p> <p>If there is a function for this Developer can set zero fee or no max wallet size for addresses (for example team wallets can trade without fee. Can cause farming)</p> <p>If there is a whitelist, some addresses may not be able to trade normally (honeypot risk).</p>
Hidden Owner Analysis		<p>No authorised hidden owner found.</p> <p>For contract with a hidden owner, developer can still manipulate the contract even if the ownership has been abandoned. Fake renounce.</p>
Retrieve Ownership Function		<p>No functions found which can retrieve ownership of the contract.</p> <p>If this function exists, it is possible for the project owner to regain ownership even after relinquishing it. Also known as fake renounce.</p>
Self Destruct Function		<p>No Self Destruct function found.</p> <p>If this function exists and is triggered, the contract will be destroyed, all functions will be unavailable, and all related assets will be erased.</p>
Specific Tax Changing Function		<p>Specific Tax Changing Functions found.</p> <p> Renounced, this function can not be used.</p> <p>If it exists, the contract owner may set a very outrageous tax rate for assigned address to block it from trading. Can assign all wallets at once!</p>
Trading Cooldown Function		<p>Trading Cooldown Function found.</p> <p> Renounced, this function can not be used.</p> <p>If there is a trading cooldown function, the user will not be able to sell the token within a certain time or block after buying. Like a temporary honeypot.</p>
Max Transaction and Holding Modify Function		<p>Max Transaction and Holding Modify function found.</p> <p> Renounced, this function can not be used.</p> <p>If there is a function for this, the maximum trading amount or maximum position can be modified. Can cause honeypot</p>
Transaction Limiting Function		<p>Transaction Limiter Function Found.</p> <p> Renounced, this function can not be used.</p> <p>The number of overall token transactions may be limited (honeypot risk)</p>

Contract Safety and Weakness

Uninitialized Local Variables (1)

This contract's local variables are not all initialized, potentially resulting in lost funds or other exploits.

Missing Arithmetic Events (2)

This contract is missing useful arithmetic events.

Missing Zero Address Validation (2)

Some functions in this contract may not appropriately check for zero addresses being used.

Incorrect Solidity Version (2)

This contract uses an unconventional or very old version of Solidity.

Public Functions Should be Declared External (5)

Some functions in this contract should be declared as external in order to save gas.

Division Before Multiplication (9)

The order of operations used may result in a loss of precision.

No compiler version inconsistencies found

No unchecked call responses found

No vulnerable self-destruct functions found

No assertion vulnerabilities found

No old solidity code found

No external delegated calls found

No external call dependency found

No vulnerable authentication calls found

No invalid character typos found

No RTL characters found

No dead code found

No risky data allocation found

No uninitialized state variables found

No uninitialized storage variables found

No vulnerable initialization functions found

No risky data handling found

No number accuracy bug found

No out-of-range number vulnerability found

No map data deletion vulnerabilities found

No tautologies or contradictions found

No faulty true/false values found

No redundant constructor calls found

No vulnerable transfers found

No vulnerable return values found

No default function responses found

No missing access control events found

No redundant true/false comparisons found

No state variables vulnerable through function calls found

No buggy low-level calls found

No expensive loops found

No bad numeric notation practices found

No missing constant declarations found


No vulnerable payable functions found

No vulnerable message values found

Detected High Severity Vulnerabilities


⚠️ Uninitialized Local Variables (1 item)


This contract's local variables are not all initialized, potentially resulting in lost funds or other exploits.

Function	Severity	Relevant Snippet
Issue Location in Code WDIStandardToken._transfer(address,address,uint256).fees (WDIStandardToken.sol#731) is a local variable never initialized	 Severity : High	uint256 fees;

⚠️ Missing Arithmetic Events (2 Items)


This contract is missing useful arithmetic events.


Function	Severity	Relevant Snippet
Issue Location in Code WDIStandardToken.updateMaxTransactionAmount(uint256) (WDIStandardToken.sol#655-659) should emit an event for: - maxAmountForTx = (maxTx * tokenInfo.totalSupply) / 10000000000000000000000 (WDIStandardToken.sol#658)	 Severity : High	function updateMaxTransactionAmount(uint256 maxTx) external onlyOwner { require(maxTx <= 100 ether && maxTx >= 0.5 ether, "TDP4"); tokenInfo.maxPercentageForTx = maxTx; maxAmountForTx = (maxTx * tokenInfo.totalSupply) / 100 ether; }

<p>Issue Location in Code</p> <pre>WDIStandardToken.updateMax WalletAmount(uint256) (WDIStandardToken.sol#649- 653) should emit an event for: - maxAmountForWallet = (maxWallet * tokenInfo.totalSupply) / 10000000000000000000000 (WDIStandardToken.sol#652)</pre>	<div>  <p>Severity : High</p> </div>	<pre>function updateMaxWalletAmount(uint256 maxWallet) external onlyOwner { require(maxWallet <= 100 ether && maxWallet >= 0.5 ether, "TDP4"); tokenInfo.maxPercentage ForWallet = maxWallet; maxAmountForWallet = (maxWallet * tokenInfo.totalSupply) / 100 ether; }</pre>
--	---	---

⚠️Missing Zero Address Validation (2 Items)



Some functions in this contract may not appropriately check for zero addresses being used.

Function	Severity	Relevant Snippet
Issue Location in Code WDIStandardToken.construct or(WDIStandardToken.TokenI nfo,uint256,address)._depl oyFeeReceiver (WDIStandardToken.sol#503) lacks a zero-check on : - deployer = _deployFeeReceiver (WDIStandardToken.sol#505)	 Severity : High	address _deployFeeReceiver

Issue Location in Code		address
WDIStandardToken.construct or(WDIStandardToken.TokenI nfo,uint256,address).swapF actory (WDIStandardToken.sol#526) lacks a zero-check on : - swapPair = IUniswapV2Factory(swapFact ory).createPair(address(th is),weth) (WDIStandardToken.sol#528)	 Severity : High	swapFactory = IUniswapV2Router02(_tokenInfo. swapRouter).factory();




Incorrect Solidity Version (2 Items)



This contract uses an unconventional or very old version of Solidity.

Function	Severity	Relevant Snippet
Pragma version^0.8.0 (WDIStandardToken.sol#5) allows old versions	 Severity : High	pragma solidity ^0.8.0;
solc-0.8.17 is not recommended for deployment	 Severity : High	

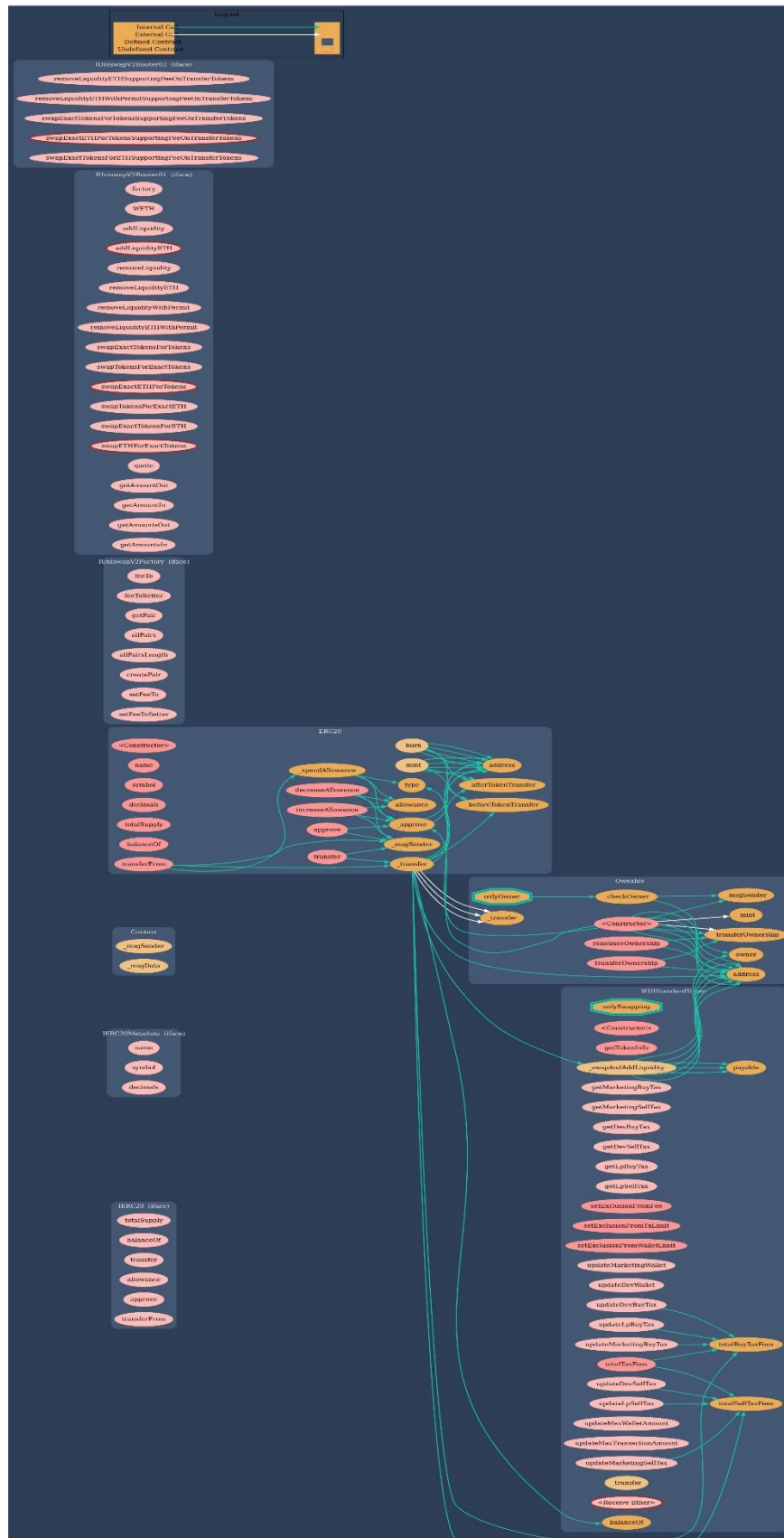
Public Functions Should be Declared External (5 Items)

Some functions in this contract should be declared as external in order to save gas.

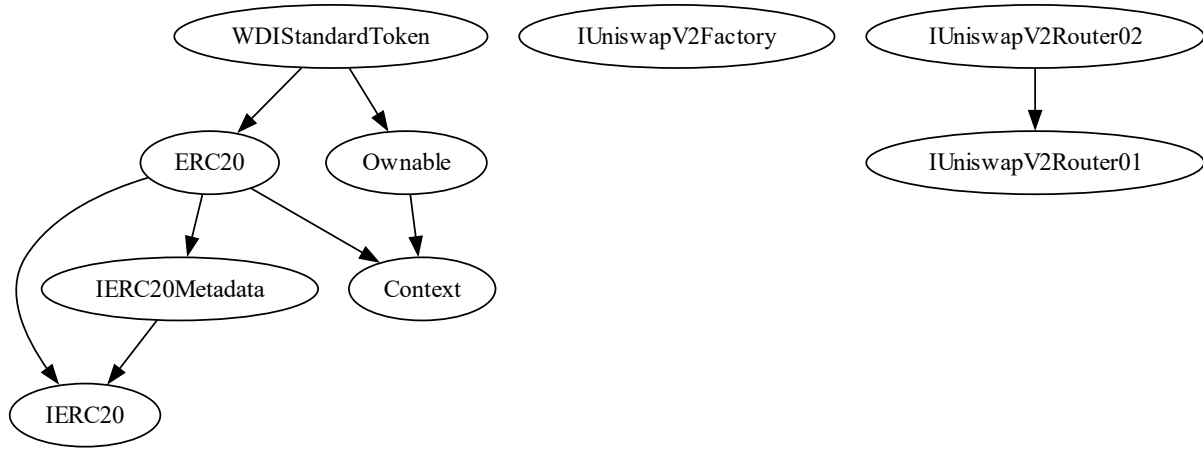
Function	Severity	Relevant Snippet
getTokenInfo() should be declared external: - WDIStandardToken.getTokenInfo() (WDIStandardToken.sol#553-555)	 Severity : High	<pre>function getTokenInfo() public view returns (TokenInfo memory _tokenInfo) { _tokenInfo = tokenInfo; }</pre>
totalTaxFees() should be declared external: - WDIStandardToken.totalTaxFees() (WDIStandardToken.sol#565-567)	 Severity : High	<pre>function totalTaxFees() public view returns (uint256) { return totalBuyTaxFees() + totalSellTaxFees(); }</pre>
setExclusionFromFee(address s,bool) should be declared external: - WDIStandardToken.setExclusionFromFee(address,bool) (WDIStandardToken.sol#593-595)	 Severity : High	<pre>function setExclusionFromFee(address account, bool value) public onlyOwner { isExcludeFromFee[account] = value; }</pre>

<p>setExclusionFromTxLimit(address,bool) should be declared external: - WDIStandardToken.setExclusionFromTxLimit(address,bool) (WDIStandardToken.sol#597-599)</p>	 Severity : High	<pre>function setExclusionFromTxLimit(address s account, bool value) public onlyOwner { isExcludeFromTxLimit[account] = value; }</pre>
<p>setExclusionFromWalletLimit(address,bool) should be declared external: - WDIStandardToken.setExclusionFromWalletLimit(address,bool) (WDIStandardToken.sol#601-603)</p>	 Severity : High	<pre>function setExclusionFromWalletLimit(address account, bool value) public onlyOwner { isExcludeFromWalletLimit[account] = value; }</pre>
















Contract Flow Graph





























Inheritance Graph



Contract Descriptions

Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
IERC20	Interface			
	totalSupply	External !		NO !
	balanceOf	External !		NO !
	transfer	External !		NO !
	allowance	External !		NO !
	approve	External !		NO !
	transferFrom	External !		NO !
Context	Implementation			
	_msgSender	Internal 		
	_msgData	Internal 		
Ownable	Implementation	Context		
		Public !		NO !
	owner	Public !		NO !
	renounceOwnership	Public !		onlyOwner
	transferOwnership	Public !		onlyOwner
	_setOwner	Private 		
SafeMath	Library			
	tryAdd	Internal 		
	trySub	Internal 		
	tryMul	Internal 		
	tryDiv	Internal 		
	tryMod	Internal 		

	add	Internal 		
	sub	Internal 		
	mul	Internal 		
	div	Internal 		
	mod	Internal 		
	sub	Internal 		
	div	Internal 		
	mod	Internal 		
BaseToken	Implementation			
StandardToken	Implementation	IERC20, Ownable, BaseToken		
		Public !		NO !
	name	Public !		NO !
	symbol	Public !		NO !
	decimals	Public !		NO !
	totalSupply	Public !		NO !
	balanceOf	Public !		NO !
	transfer	Public !		NO !
	allowance	Public !		NO !
	approve	Public !		NO !
	transferFrom	Public !		NO !
	increaseAllowance	Public !		NO !
	decreaseAllowance	Public !		NO !
	_transfer	Internal 		
	_mint	Internal 		
	_burn	Internal 		

	<code>_approve</code>	Internal 		
	<code>_setupDecimals</code>	Internal 		
	<code>_beforeTokenTransfer</code>	Internal 		



Function can
modify state



Function
is payable

Source

File Name

SHA-1 Hash

c:\Solidity\bigcats.sol 119b7f3562c451057171fee8b8602b804e1febd1