SKELETON ECOSYSTEM

SMART CONTRACT AUDIT

BIG CATS BSC

BEP20

0xDE08DC969aF9D2024C8983287d2C2f74E1f46396

Table of Contents

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% safaty 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	BIG CATS BSC
Ticker/Simbol	(BCAT)
Blockchain	56 Binance Smart Chain BEP20
Contract Address	0xDE08DC969aF9D2024C8983287d2C2f74E1f46396
Creator Address	0x40187Bf465967a06aB6aAe59C985C8E31FfADfBe
Current Owner Address	Renounced
Contract Explorer	https://bscscan.com/address/0xde08dc969af9d2024c8983287d2c2f74e1f46396
Compiler Version	v0.8.4+commit.c7e474f2
License	MIT License
Optimisation	Yes with 200 Runs
Total Supply	1,000,000,000,000,000 BCAT
Decimals	17

Creation/Audit

Contract Deployed	31-Jul-23 17:07:05 UTC
Audit Created	09-Aug-23 15:00:00 UTC
Audit Update	V 0.1

Verified Socials

Website	https://bigcats.com
Telegram	https://t.me/bigcatsupdate
X	https://twitter.com/bigcatsbsc

BIG Cats BSC

Contract Function Analysis

Pass Attention Item Risky Item

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	0%	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	0%	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
Liqudity Status	✅	Locked on 08.08.2023: 89% for 23days Note! Initial liqudity 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 modifiy 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	✅	No Balance Modifier function found. 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. ⚠ If contract is renounced this function still can be used as auto self Destruct
Whitelist Function	✅	No Whitelist Function Found. If there is a function for this Developer can set zero fee or no max wallet size for adresses (for example team wallets can trade without fee. Can cause farming) If there is a whitelist, some addresses may not be able to trade normally (honeypot risk).
Hidden Owner Analysis	✅	No authorised hidden owner found. For contract with a hidden owner, developer can still manipulate the contract even if the ownership has been abandoned. Fake renounce.
Retrieve Ownership Function	✅	No functions found which can retrieve ownership of the contract. If this function exists, it is possible for the project owner to regain ownership even after relinquishing it. Also known as fake renounce.
Self Destruct Function	✅	No Self Destruct function found. If this function exists and is triggered, the contract will be destroyed, all functions will be unavailable, and all related assets will be erased.
Specific Tax Changing Function	⚠	Specific Tax Changing Functions found. ✅ Renounced, this function can not be used. 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!
Trading Cooldown Function	⚠	Trading Cooldown Function found. ✅ Renounced, this function can not be used. 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.
Max Transaction and Holding Modify Function	⚠	Max Transaction and Holding Modify function found. ✅ Renounced, this function can not be used. If there is a function for this, the maximum trading amount or maximum position can be modified. Can cause honeypot

Transaction Limiting Function	⚠	Transaction Limiter Function Found. ✅ Renounced, this function can not be used. The number of overall token transactions may be limited (honeypot risk)

Domain Analysis of https://bigcats.com

Registrar: Key-Systems GmbH

IANA ID: 269

URL: https://www.hostinger.com/contacts,http://www.key-systems.net

Whois Server: whois.rrpproxy.net

Created on: 2023-05-06

Expires on: 2024-05-06

Updated on: 2023-05-06

IP Address: 86.38.202.205 - 211 other sites hosted on this server

IP Location: United States - Maine - Hallowell - Sc Lithuanian Radio And Tv Center

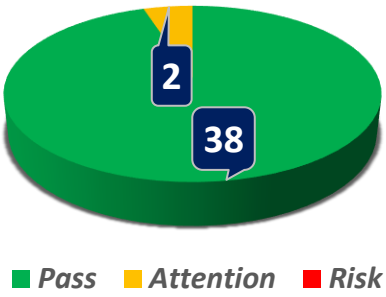
ASN: United States AS47583 AS-HOSTINGER Hostinger International Limited, CY (registered Apr 04, 2011)

Contract Safety and Weakness

Pass

Attention Item

Risky Item



Zero Address Validation	⚠
Public Functions	⚠
Compiler version inconsistencies	✓
Unchecked call responses	✓
Vulnerable self-destruct functions	✓
Assertion vulnerabilities	✓
Old solidity code	✓
External delegated calls	✓
External call dependency	✓
Vulnerable authentication calls	✓
Invalid character typos	✓
RTL characters	✓
Dead code	✓
Risky data allocation	✓
Uninitialized state variables	✓
Uninitialized storage variables	✓
Vulnerable initialization functions	✓
Risky data handling	✓
Number accuracy bug	✓
Out-of-range number vulnerability	✓

Map data deletion vulnerabilities	✅
Tautologies or contradictions	✅
Faulty true/false values	✅
Innacurate divisions	✅
Redundant constructor calls	✅
Vulnerable transfers	✅
Vulnerable return values	✅
Uninitialized local variables	✅
Default function responses	✅
Missing arithmetic events	✅
Missing access control events	✅
Redundant true/false comparisons	✅
State variables vulnerable through function calls	✅
Buggy low-level calls	✅
Invalid solidity versions	✅
Expensive loops	✅
Bad numeric notation practices	✅
Missing constant declarations	✅
Vulnerable payable functions	✅
Vulnerable message values	✅

Detected Vulnerabilities

Missing Zero Address Validation (1 Detected Item)

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

Function	Severity	Relevant Snippet
Issue Location in Code StandardToken.constructor(string,string,uint8,uint256,address,uint256).serviceFeeReceiver_ (StandardToken.sol#467) lacks a zero-check on : - address(serviceFeeReceiver_).transfer(serviceFee_) (StandardToken.sol#477)	Severity: Moderate	address serviceFeeReceiver_,

⚠ Public Functions Should be Declared External (11 Detected Items)

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

Function	Severity	Relevant Snippet

name() should be declared external: - StandardToken.name() (StandardToken.sol#483-485)	Severity: High	function name() public view virtual returns (string memory) {
		return _name;
		}

symbol() should be declared external: - StandardToken.symbol() (StandardToken.sol#491-493)	Severity: High	function symbol() public view virtual returns (string memory) {
		return _symbol;
		}

decimals() should be declared external:
- StandardToken.decimals() (StandardToken.sol#508-510)
	Severity: High	function decimals() public view virtual returns (uint8) {
		return _decimals;
		}

totalSupply() should be declared external: - StandardToken.totalSupply() (StandardToken.sol#515-517)	Severity: High	function totalSupply() public view virtual override returns (uint256) {
		return _totalSupply;
		}

balanceOf(address) should be declared external: - StandardToken.balanceOf(address) (StandardToken.sol#522-530)	Severity: High	function balanceOf(address account) public view virtual override returns (uint256) return _balances[account]; }

Issue Location in Code
 	
transfer(address,uint256) should be declared external:
- StandardToken.transfer(address,uint256) (StandardToken.sol#540-548)
	
	Severity: High
		function transfer(address recipient, uint256 amount)
		public
		virtual
		override
		returns (bool)
		_transfer(_msgSender(), recipient, amount);
		return true;
		}

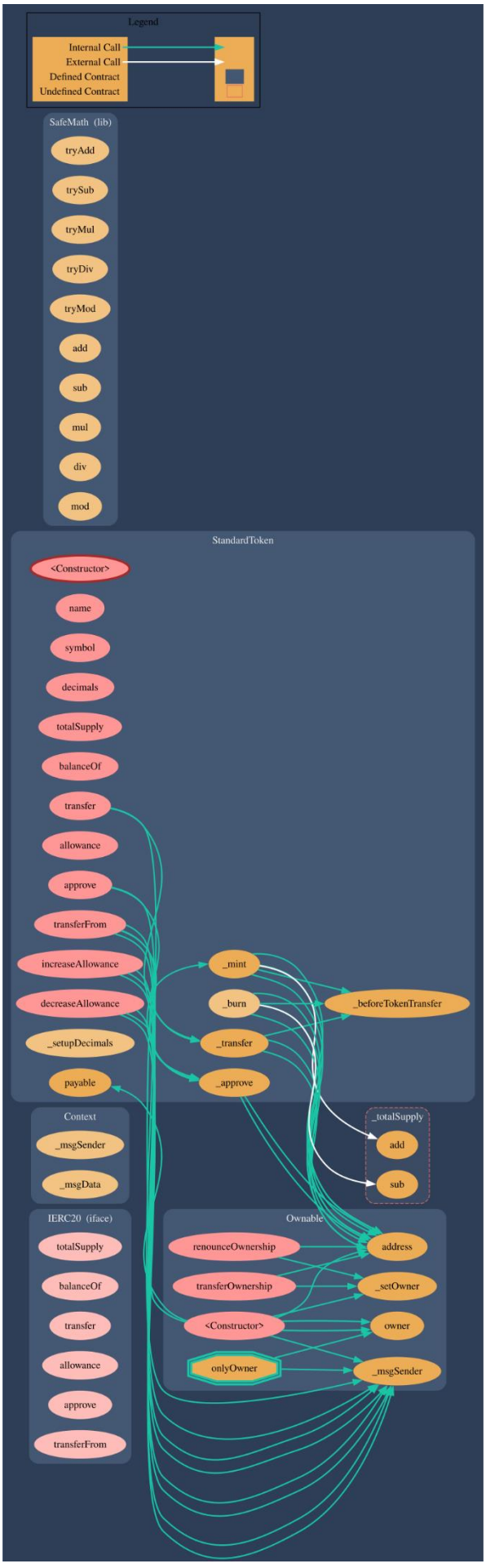
allowance(address,address) should be declared external: - StandardToken.allowance(address, address) (StandardToken.sol#553-561)	 Severity: High	function allowance(address owner, address spender) public view virtual override returns (uint256) return _allowances[owner][spender]; }

Issue Location in Code
 	
approve(address,uint256) should be declared external:
- StandardToken.approve(address,uint256) (StandardToken.sol#570-578)
	
	Severity: High
		function approve(address spender, uint256 amount)
		public
		virtual
		override
		returns (bool)
		_approve(_msgSender(), spender, amount);
		return true;
		}

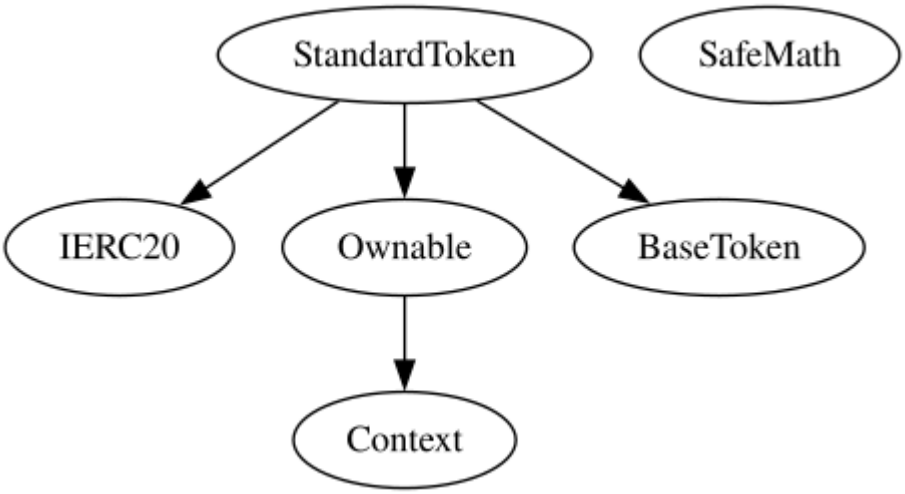
Issue Location in Code
 	
transferFrom(address,address,uint256) should be declared external:
- StandardToken.transferFrom(address, address,uint256) (StandardToken.sol#593-608)
	
	Severity: High
		function transferFrom(
		address sender,
		address recipient,
		uint256 amount
		) public virtual override returns (bool) {
		_transfer(sender, recipient, amount);
		_approve(
		sender,
		_msgSender(),
		
		_allowances[sender][_msgSender()].sub(
		amount,
		"ERC20: transfer amount exceeds allowance"
		)
		);
		return true;}

Issue Location in Code increaseAllowance(address,uint256) should be declared external: - StandardToken.increaseAllowance(address,uint256) (StandardToken.sol#622-633)	 Severity: High	function increaseAllowance(address spender, uint256 addedValue) public virtual returns (bool) _approve(_msgSender(), spender, _allowances[_msgSender()][spender].add(addedValue)); return true; }
Issue Location in Code decreaseAllowance(address,uint256) should be declared external: - StandardToken.decreaseAllowance(address,uint256) (StandardToken.sol#649-663)	 Severity: High	function decreaseAllowance(address spender, uint256 subtractedValue) public virtual returns (bool) _approve(_msgSender(), spender, _allowances[_msgSender()][spender].sub(subtractedValue, "ERC20: decreased allowance below zero")); return true; }

Contract Flow Chart



Inheritance Chart



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 🔒	🛑	
	_approve	Internal 🔒	🛑	
	_setupDecimals	Internal 🔒	🛑	
	_beforeTokenTransfer	Internal 🔒	🛑	

🛑 Function can modify state

💵 Function is payable