Cyberscope

Audit Report

Rocket Snail

March 2024

Network	BSC
Address	0xdf0fed58f0e2ba647ae1eabdfae817e0c73ee6f3
Audited by	© cyberscope

Analysis

Critical Medium Minor / Informative Pass

Severity	Code	Description	Status
●	ST	Stops Transactions	Passed
●	OTUT	Transfers User's Tokens	Passed
●	ELFM	Exceeds Fees Limit	Passed
●	MT	Mints Tokens	Passed
●	BT	Burns Tokens	Passed
●	BC	Blacklists Addresses	Passed

Diagnostics

Critical Medium Minor / Informative

Severity	Code	Description	Status
●	RAC	Redundant Address Casting	Unresolved

Table of Contents

Review

Contract Name	BurnableTeamToken
Compiler Version	v0.6.12+commit.27d51765
Optimization	200 runs
Explorer	https://bscscan.com/address/0xdf0fed58f0e2ba647ae1eabdfae817e0c73ee6f3
Address	0xdf0fed58f0e2ba647ae1eabdfae817e0c73ee6f3
Network	BSC
Symbol	RSNAIL
Decimals	18
Total Supply	1,000,000,000
Badge Eligibility	Yes

Audit Updates

Initial Audit	13 Mar 2024

Source Files

Filename	SHA256
contracts/TeamToken.sol	75ea52da4fcc79984c5ec91038e7f667c2a8927493e7d9b235bcba98ae84cc0e
contracts/SafeMath.sol	c12e80c632ffe9759da012ded8a255048b87d4ee6d7156c7aa9f7e98e4b9a6b6

contracts/IERC20.sol	397534ef6f97a7fa11089b8f05cf5383749067352d06c0e39b880fd08743489a
contracts/ERC20.sol	2801451a608b8512b9dcfff8d81a61805dd7387ea90e74c09478fac336979263
contracts/Context.sol	6a25312554a817075fbe85e3c57f5e0ecad5b0bac4303bd3967ce9680e71a2af
contracts/BurnableToken.sol	d3ea637d1b59d04ad5af29d2a6de57d7bd24cd3142c3d6e95b2446bd60e2787f
contracts/BurnableTeamToken.sol	22816440aa73d81ea7ba86ec80377544ae3667d0309eb137f66eba17a24759bc

Findings Breakdown

Critical	0
Medium	0
Minor / Informative	1

Severity	Unresolved	Acknowledged	Resolved	Other
Critical	0	0	0	0
Medium	0	0	0	0
Minor / Informative	1	0	0	0

RAC - Redundant Address Casting

Criticality	Minor / Informative
Location	contracts/TeamToken.sol#L15
Status	Unresolved

Description

The contract is implementing the `checkIsAddressValid` modifier that is used to validate Ethereum addresses passed as arguments to the contract's constructor. The modifier attempts to cast `ethAddress` to an `address` type and compare it to its original value. Given that `ethAddress` is already declared and passed as an address type through the constructor parameters, this casting operation is redundant. The Solidity type system ensures that `ethAddress` is a valid address by definition, making this check unnecessary and potentially confusing.

```
modifier checkIsAddressValid(address ethAddress)
{
    require(ethAddress != address(0), "[Validation] invalid
address");
    require(ethAddress == address(ethAddress), "[Validation]
invalid address");
    _;
}
constructor(
    ...
    address owner,
    address feeWallet
) public checkIsAddressValid(owner)
checkIsAddressValid(feeWallet) ERC20(name, symbol) {
    ...
```

Recommendation

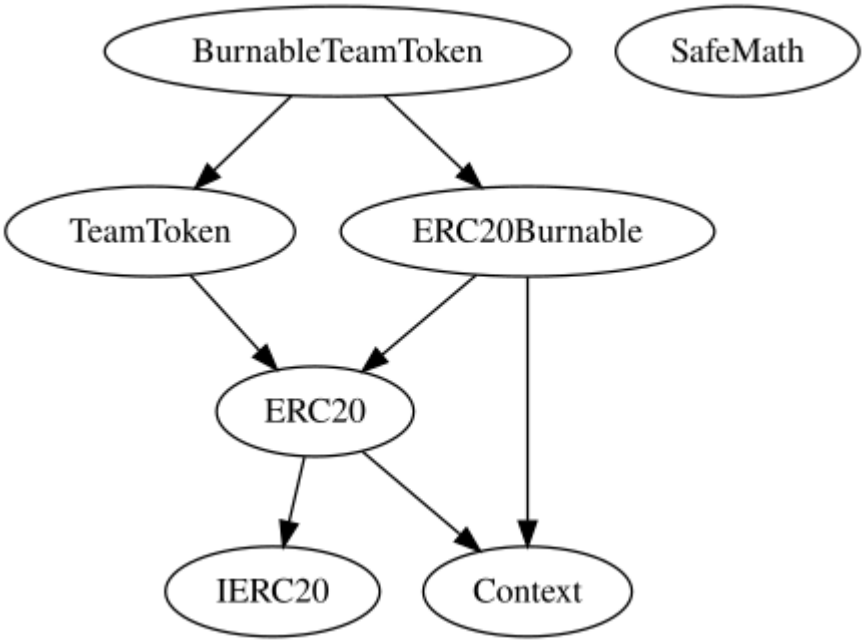
It is recommended to remove the redundant `address` casting check from the `checkIsAddressValid` modifier. This simplification will not only make the code cleaner and more efficient by eliminating an unnecessary operation but also enhance readability

and maintainability. Retaining the check for the zero address is sufficient for ensuring that the address is not the default, zero address, which is a common and meaningful validation requirement in smart contract development. Simplifying the validation logic in this manner adheres to best practices for smart contract development by focusing on clarity and efficiency.

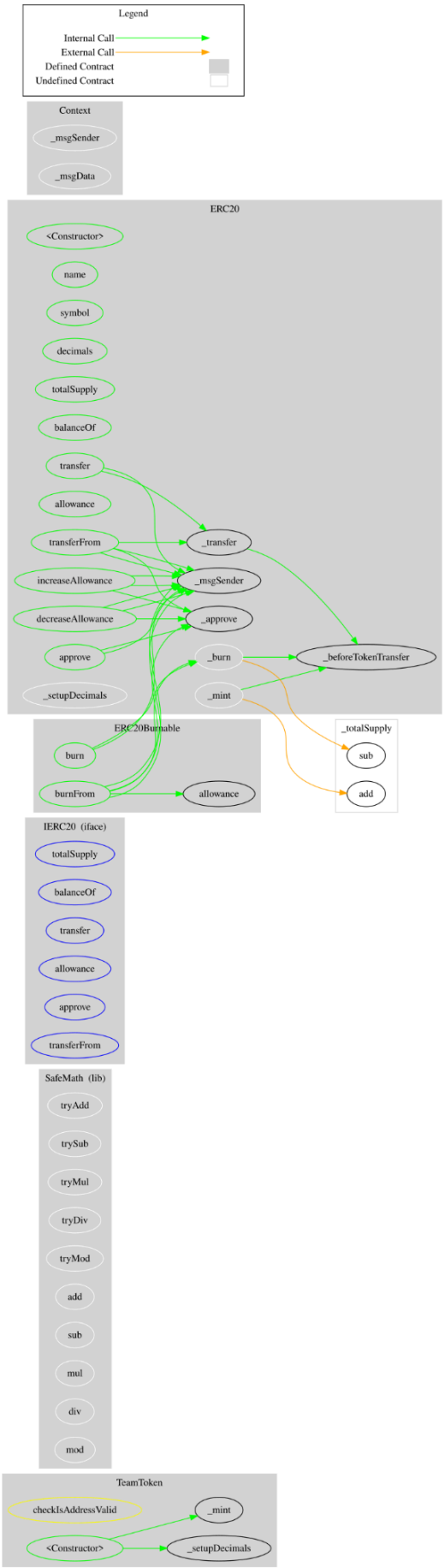
Functions Analysis

Contract	Type	Bases		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
TeamToken	Implementation	ERC20		
		Public	✓	checkIsAddress Valid checkIsAddress Valid ERC20
ERC20Burnable	Implementation	Context, ERC20		
	burn	Public	✓	-
	burnFrom	Public	✓	-
BurnableTeamToken	Implementation	TeamToken, ERC20Burnable		
		Public	✓	TeamToken

Inheritance Graph



Flow Graph


Legend
Internal Call
External Call
Defined Contract
Undefined Contract
Context
_msgSender
_msgData
ERC20
<Constructor>
name
symbol
decimals
totalSupply
balanceOf
transfer
allowance
transferFrom
increaseAllowance
decreaseAllowance
approve
_setupDecimals
_transfer
_msgSender
_approve
_burn
_mint
_beforeTokenTransfer
ERC20Burnable
burn
burnFrom
allowance
_totalSupply
sub
add
IERC20 (iface)
totalSupply
balanceOf
transfer
allowance
approve
transferFrom
SafeMath (lib)
tryAdd
trySub
tryMul
tryDiv
tryMod
add
sub
mul
div
mod
TeamToken
checkIsAddressValid
<Constructor>
_mint
_setupDecimals

Summary

Rocket Snail contract implements a token mechanism. This audit investigates security issues, business logic concerns and potential improvements. Rocket Snail is an interesting project that has a friendly and growing community. The Smart Contract analysis reported no compiler error or critical issues. The contract Owner can access some admin functions that can not be used in a malicious way to disturb the users' transactions.

Disclaimer

The information provided in this report does not constitute investment, financial or trading advice and you should not treat any of the document's content as such. This report may not be transmitted, disclosed, referred to or relied upon by any person for any purposes nor may copies be delivered to any other person other than the Company without Cyberscope's prior written consent. This report is not nor should be considered an "endorsement" or "disapproval" of any particular project or team. This report is not nor should be regarded as an indication of the economics or value of any "product" or "asset" created by any team or project that contracts Cyberscope to perform a security assessment. This document does not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors' business, business model or legal compliance. This report should not be used in any way to make decisions around investment or involvement with any particular project. This report represents an extensive assessment 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 Cyberscope's position is that each company and individual are responsible for their own due diligence and continuous security Cyberscope'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. The assessment services provided by Cyberscope are subject to dependencies and are under continuing development. You agree that your access and/or use including but not limited to any services reports and materials will be at your sole risk on an as-is where-is and as-available basis Cryptographic tokens are emergent technologies and carry with them high levels of technical risk and uncertainty. The assessment reports could include false positives false negatives and other unpredictable results. The services may access and depend upon multiple layers of third parties.

About Cyberscope

Cyberscope is a blockchain cybersecurity company that was founded with the vision to make web3.0 a safer place for investors and developers. Since its launch, it has worked with thousands of projects and is estimated to have secured tens of millions of investors' funds.

Cyberscope is one of the leading smart contract audit firms in the crypto space and has built a high-profile network of clients and partners.

The Cyberscope team

https://www.cyberscope.io