Cyberscope

Audit Report

Bananace

May 2023

Network	BSC
Address	0x6C8C79c1e310C879234E4Fd0e943A19e0524265f
Audited by	© cyberscope

Table of Contents

Review

Contract Name	ERC20TokenOB
Compiler Version	v0.8.17+commit.8df45f5f
Optimization	200 runs
Explorer	https://bscscan.com/address/0x21d714527f1e4f62abb7f68bcd7c94e94d8121f9
Address	0x21d714527f1e4f62abb7f68bcd7c94e94d8121f9
Network	BSC
Symbol	NANA
Decimals	18
Total Supply	696,969,696,969,696

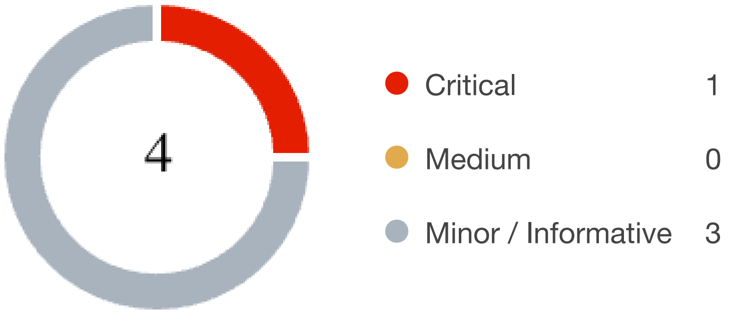
Audit Updates

Initial Audit	12 May 2023
Corrected Phase 2	14 May 2023

Source Files

Filename	SHA256
@openzeppelin/contracts/access/Ownable.sol	9353af89436556f7ba8abb3f37a6677249aa4df6024fbfaa94f79ab2f44f3231
@openzeppelin/contracts/token/ERC20/ERC20.sol	bce14c3fd3b1a668529e375f6b70ffdf9cef8c4e410ae99608be5964d98fa701
@openzeppelin/contracts/token/ERC20/extensions/ERC20Burnable.sol	0344809a1044e11ece2401b4f7288f414ea41fa9d1dad24143c84b737c9fc02e
@openzeppelin/contracts/token/ERC20/extensions/IERC20Metadata.sol	af5c8a77965cc82c33b7ff844deb9826166689e55dc037a7f2f790d057811990
@openzeppelin/contracts/token/ERC20/IERC20.sol	94f23e4af51a18c2269b355b8c7cf4db8003d075c9c541019eb8dcf4122864d5
@openzeppelin/contracts/utils/Context.sol	1458c260d010a08e4c20a4a517882259a23a4baa0b5bd9add9fb6d6a1549814a
contracts/interfaces/IBananaAntiBot.sol	be576cc14d95e13dda2706091de23ed37733ff6ab7086c0661635afcd01898bb
contracts/tokens/ERC20TokenOB.sol	1c7765984d00cd68ebf57bd75ebb285c8a6719997a13822d5afd56e1b19fbe8e

Findings Breakdown



Severity	Unresolved	Acknowledged	Resolved	Other
● Critical	1	0	0	0
● Medium	0	0	0	0
● Minor / Informative	3	0	0	0

Analysis

🔴 Critical 🟠 Medium ⚪ Minor / Informative 🔵 Pass

Severity	Code	Description	Status
🔴	ST	Stops Transactions	Unresolved
🔵	OCTD	Transfers Contract's Tokens	Passed
🔵	OTUT	Transfers User's Tokens	Passed
🔵	ELFM	Exceeds Fees Limit	Passed
🔵	ULTW	Transfers Liquidity to Team Wallet	Passed
🔵	MT	Mints Tokens	Passed
🔵	BT	Burns Tokens	Passed
🔵	BC	Blacklists Addresses	Passed

ST - Stops Transactions

Criticality	Critical
Location	contracts/tokens/ERC20TokenOB.sol#L53
Status	Unresolved

Description

The contract uses an external contract in order to determine the transaction's flow. The external contract is untrusted. As a result, it may produce security issues and harm the transactions.

```
if(from != owner() && to != owner()) {
  require(amount <= _maxTxAmount, "Transfer amount exceeds the
maxTxAmount.");
}
```

Recommendation

The contract should use a trusted external source. A trusted source could be either a commonly recognized or an audited contract. The pointing addresses should not be able to change after the initialization.

Diagnostics

● Critical ● Medium ● Minor / Informative

Severity	Code	Description	Status
●	RDE	Redundant Decimals Extension	Unresolved
●	RMO	Redundant Mint Override	Unresolved
●	L19	Stable Compiler Version	Unresolved

RDE - Redundant Decimals Extension

Criticality	Minor / Informative
Location	contracts/tokens/ERC20TokenOB.sol#L29
Status	Unresolved

Description

The contract implements the Openzeplin ERC20 standard. According to the ERC20 standard, the decimals are 18. The contract overrides the decimals method. This method returns the same number as the ERC20 standard. As a result, the override is redundant.

```
function decimals() public view virtual override returns (uint8) {
    return DECIMALS;
}
```

Recommendation

The team is advised to remove the decimals override since it will produce the same result.

RMO - Redundant Mint Override

Criticality	Minor / Informative
Location	contracts/tokens/ERC20TokenOB.sol#L40
Status	Unresolved

Description

The contract executes the mint method once in the constructor. The `mint()` method is overridden by the contract to allow only non-zero mints. Since the mint method is called once in the constructor, then the override of the mint method is redundant.

```
function _mint(address account, uint256 amount) internal virtual
override {
    if (amount > 0) {
        super._mint(account, amount);
    }
}
```

Recommendation

The team is advised to remove the override of the `mint()` method since it will produce the same result.

L19 - Stable Compiler Version

Criticality	Minor / Informative
Location	contracts/tokens/ERC20TokenOB.sol#L3
Status	Unresolved

Description

The `^` symbol indicates that any version of Solidity that is compatible with the specified version (i.e., any version that is a higher minor or patch version) can be used to compile the contract. The version lock is a mechanism that allows the author to specify a minimum version of the Solidity compiler that must be used to compile the contract code. This is useful because it ensures that the contract will be compiled using a version of the compiler that is known to be compatible with the code.

```
pragma solidity ^0.8.17;
```

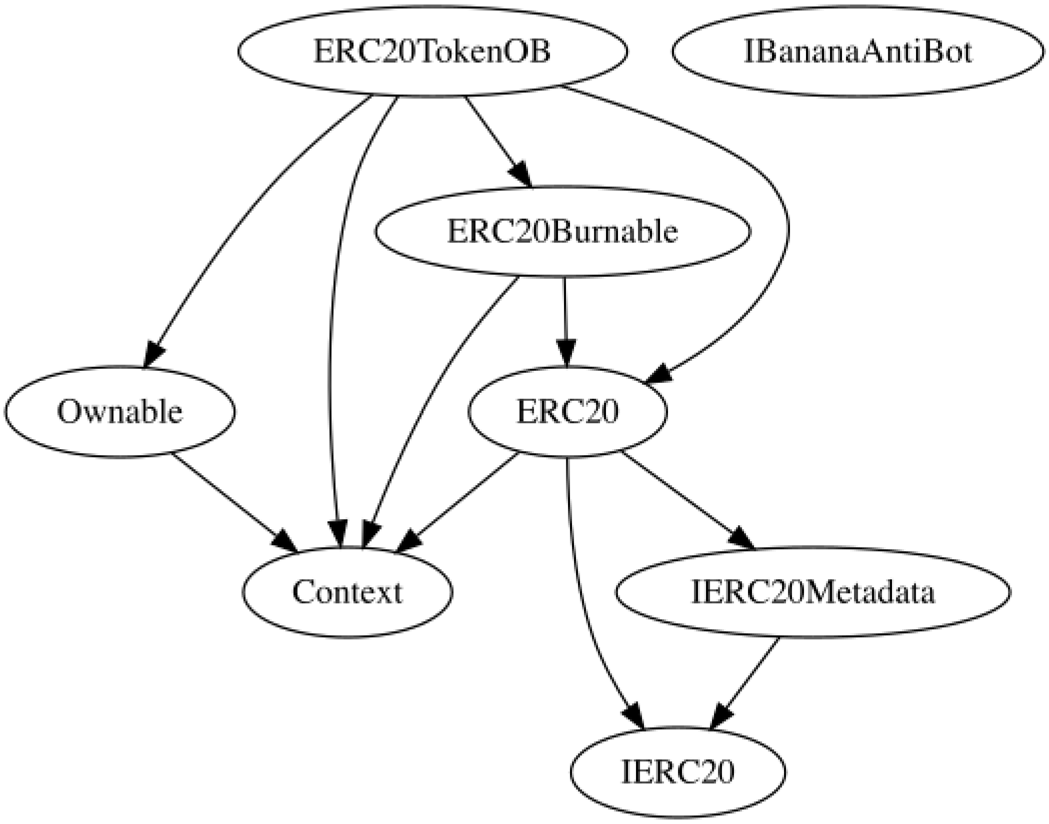
Recommendation

The team is advised to lock the pragma to ensure the stability of the codebase. The locked pragma version ensures that the contract will not be deployed with an unexpected version. An unexpected version may produce vulnerabilities and undiscovered bugs. The compiler should be configured to the lowest version that provides all the required functionality for the codebase. As a result, the project will be compiled in a well-tested LTS (Long Term Support) environment.

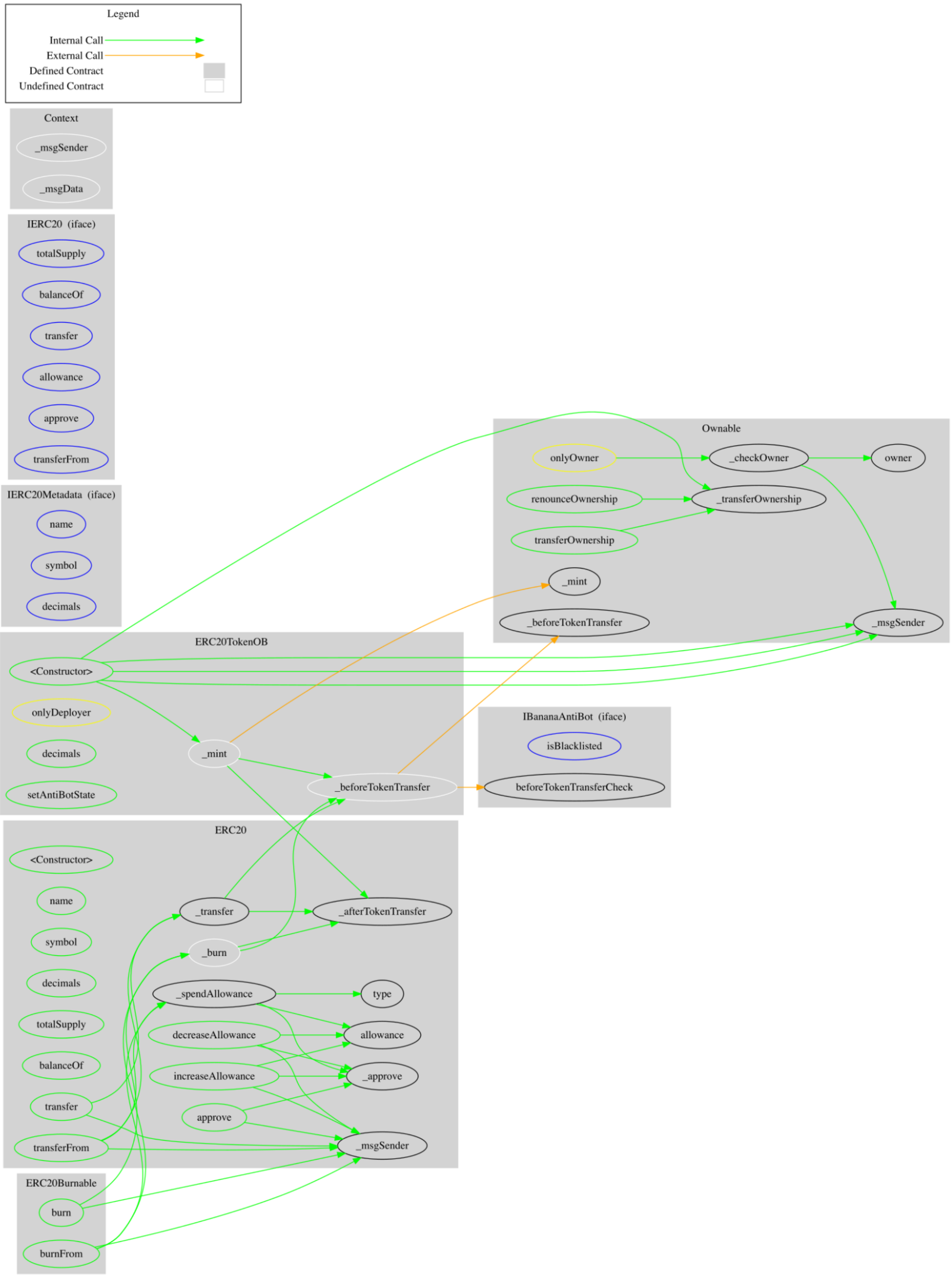
Functions Analysis

Contract	Type	Bases		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
ERC20TokenOB	Implementation	Context, ERC20, ERC20Burnable, Ownable		
		Public	✓	ERC20
	decimals	Public		-
	setAntiBotState	Public	✓	onlyDeployer
	_mint	Internal	✓	
	_beforeTokenTransfer	Internal	✓	

Inheritance Graph



Flow Graph


Legend
Internal Call
External Call
Defined Contract
Undefined Contract
Context
_msgSender
_msgData
IERC20 (iface)
totalSupply
balanceOf
transfer
allowance
approve
transferFrom
IERC20Metadata (iface)
name
symbol
decimals
ERC20TokenOB
<Constructor>
onlyDeployer
decimals
setAntiBotState
_mint
_beforeTokenTransfer
Ownable
onlyOwner
_checkOwner
owner
renounceOwnership
_transferOwnership
transferOwnership
_mint
_beforeTokenTransfer
_msgSender
IBananaAntiBot (iface)
isBlacklisted
beforeTokenTransferCheck
ERC20
<Constructor>
name
symbol
decimals
totalSupply
balanceOf
transfer
transferFrom
_transfer
_afterTokenTransfer
_burn
_spendAllowance
type
decreaseAllowance
allowance
increaseAllowance
_approve
approve
_msgSender
ERC20Burnable
burn
burnFrom

Summary

Bananace contract implements a token mechanism. This audit investigates security issues, business logic concerns, and potential improvements. The contract uses an external untrusted source in order to determine the transaction flow. The team is advised to use a commonly recognized or audited contracts for external interactions.

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