Cyberscope

Audit Report

FrontFanz

November 2023

Network	MATIC
Address	0xb58458c52b6511dc723D7d6F3Be8c36D7383b4A8
Network	ETH
Address	0x3aC81633A291f342b62e7de5d00eb02924032e06
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
	L19	Stable Compiler Version	Unresolved

Table of Contents

Review

Contract Name	Token
Compiler Version	v0.8.18+commit.87f61d96
Optimization	200 runs
Explorer	https://polygonscan.com/address/0xb58458c52b6511dc723d7d6f3be8c36d7383b4a8
Address	0xb8c601785d38067a8c0141cb2ba2f0c7b060075b
Network	MATIC
Explorer	https://etherscan.io/address/0x3ac81633a291f342b62e7de5d00eb02924032e06
Address	0xa1176893d3AB91a5Ef86022FE3C7CA556AEb6339
Network	ETH
Symbol	FANX
Decimals	18
Total Supply	1,000,000,000

Audit Updates

Initial Audit	05 Nov 2022 https://github.com/cyberscope-io/audits/blob/main/fanz/v1/audit.pdf
Corrected Phase 2	16 Mar 2023 https://github.com/cyberscope-io/audits/blob/main/fanz/v2/audit.pdf

Corrected Phase 3	28 Apr 2023 https://github.com/cyberscope-io/audits/blob/main/fanz/v3/audit.pdf
Corrected Phase 4	09 Oct 2023 https://github.com/cyberscope-io/audits/blob/main/fanz/v4/audit.pdf
Corrected Phase 5	03 Nov 2023

Source Files

Filename	SHA256
Token.sol	ca7d19978a9fd9cf12d0beefc7a413fcf84d569bffb12b7edf8a037d5bc83308

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

L19 - Stable Compiler Version

Criticality	Minor / Informative
Location	Token.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 ^0.8.0;
```

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.

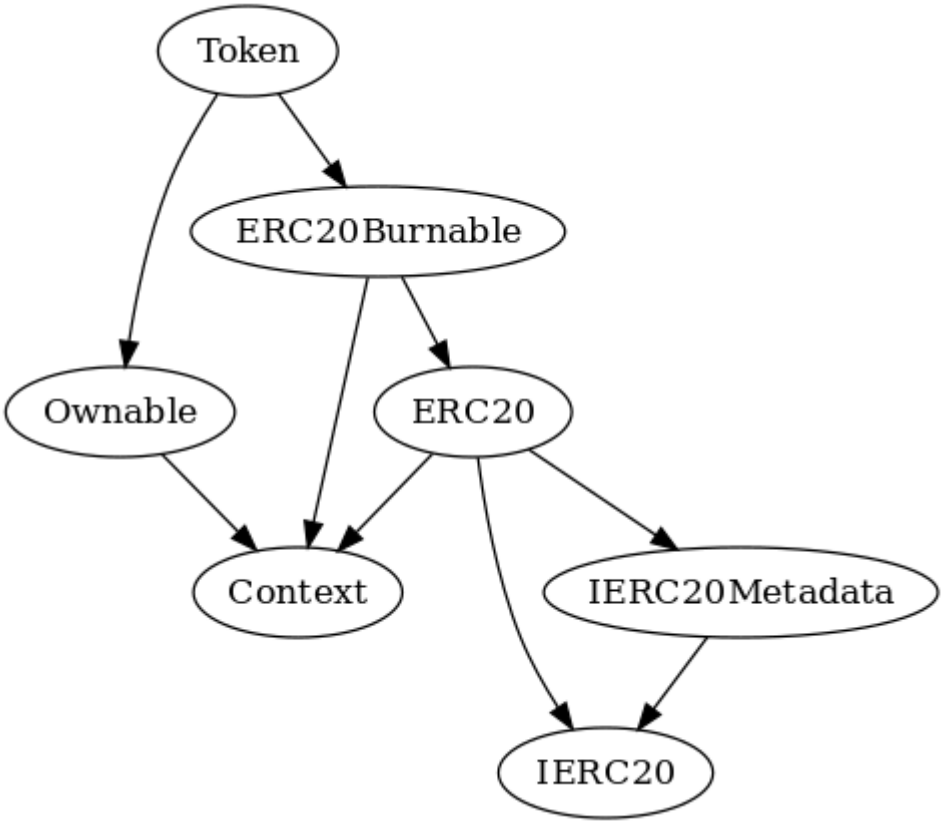
Functions Analysis

Contract	Type	Bases		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
IERC20Metadata	Interface	IERC20		
	name	External		-
	symbol	External		-
	decimals	External		-
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
Ownable	Implementation	Context		

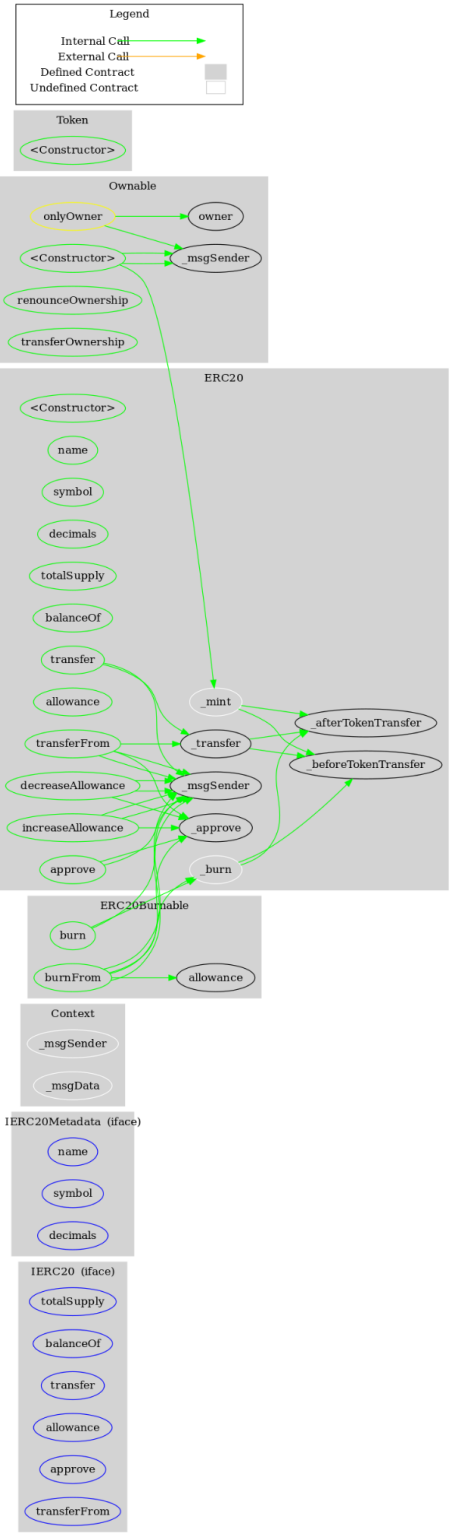
		Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
ERC20	Implementation	Context, IERC20, IERC20Metadata		
		Public	✓	-
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-
	transferFrom	Public	✓	-
	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	_transfer	Internal	✓	
	_mint	Internal	✓	
	_burn	Internal	✓	
	_approve	Internal	✓	

	_beforeTokenTransfer	Internal	✓	
	_afterTokenTransfer	Internal	✓	
ERC20Burnable	Implementation	Context, ERC20		
	burn	Public	✓	-
	burnFrom	Public	✓	-
Token	Implementation	ERC20Burnable, Ownable		
		Public	✓	ERC20

Inheritance Graph



Flow Graph


Legend
Internal Call
External Call
Defined Contract
Undefined Contract
Token
<Constructor>
Ownable
onlyOwner
owner
<Constructor>
msgSender
renounceOwnership
transferOwnership
ERC20
<Constructor>
name
symbol
decimals
totalSupply
balanceOf
transfer
allowance
_mint
_afterTokenTransfer
transferFrom
_transfer
_beforeTokenTransfer
decreaseAllowance
msgSender
increaseAllowance
_approve
approve
_burn
ERC20Burnable
burn
burnFrom
allowance
Context
_msgSender
_msgData
IERC20Metadata (iface)
name
symbol
decimals
IERC20 (iface)
totalSupply
balanceOf
transfer
allowance
approve
transferFrom

Summary

FrontFanz contract implements a token mechanism. This audit investigates security issues, business logic concerns, and potential improvements. FrontFanz is an interesting project that has a friendly and growing community. The Smart Contract analysis reported no compiler errors 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