Cyberscope

Audit Report

PayMe Crowdsale

December 2022

Github	https://github.com/payMeQuiz/payMe-Project
Commit	0dc29331c643bfaa1e71a51b8605ae6f6f8819b5
Audited by	© cyberscope

Table of Contents

Contract Review

Contract Name	payMETokenCrowdsale
Compiler Version	v0.8.9+commit.e5eed63a
Github	https://github.com/payMeQuiz/payMe-Project
Commit	0dc29331c643bfaa1e71a51b8605ae6f6f8819b5
Testing Deploy	https://testnet.bscscan.com/token/0x7fC5C8D4b416d1FAbF0c6A313523599B0045FDdD
Domain	https://payme.games

Audit Updates

Initial Audit	17th October 2022 https://github.com/cyberscope-io/audits/blob/main/payme/v1/paymeTokenCrowdsale.pdf
Corrected Phase 1	9th November 2022 https://github.com/cyberscope-io/audits/blob/main/payme/v2/paymeTokenCrowdsale.pdf
Corrected Phase 2	8th December 2022

Source Files

Filename	SHA256
@openzeppelin/contracts-upgradeable/access/OwnableUpgradeable.sol	da66c17044345dc892d85bd7ddc9745d25df0b3dacfba8f84eb87c60d6e40fe3
@openzeppelin/contracts-upgradeable/proxy/utils/Initializable.sol	74def996fd6faf32f13ab9cacfc71d57400177de340fe5d5d7c6e805dfbab3bd
@openzeppelin/contracts-upgradeable/security/ReentrancyGuardUpgradeable.sol	fcfc8be28dd0e725a6c61648b3c7a422f0e668ad2eb83c39c3c07f590846523a
@openzeppelin/contracts-upgradeable/token/ERC20/extensions/draft-IERC20PermitUpgradeable.sol	b97515a88e75c313eacf0a27c9439ef371d86d4c2730d3b13076640942f813df
@openzeppelin/contracts-upgradeable/token/ERC20/IERC20Upgradeable.sol	4e09a7479aa3e7c313f8fc141c4c8fc04e0abfeb8754615ef7d78ec94c298b07
@openzeppelin/contracts-upgradeable/token/ERC20/utils/SafeERC20Upgradeable.sol	45b47dd617d02875a7e6c896d1842ff9d8362ab15b8180645f3f4b180d4f028f

@openzeppelin/contracts-upgradeable/utils/AddressUpgradeable.sol	1d7d481b79fd54d957c9a0696f6227f7799fec01d8ba41f5c130a7cc6b4eddc9
@openzeppelin/contracts-upgradeable/utils/ContextUpgradeable.sol	5fb301961e45cb482fe4e05646d2f529aa449fe0e90c6671475d6a32356fa2d4
@openzeppelin/contracts-upgradeable/utils/math/MathUpgradeable.sol	158a0316fa289fad12c2ca764449e43e6724fb79c58fc438508d116f9af46b39
@openzeppelin/contracts-upgradeable/utils/math/SafeMathUpgradeable.sol	4039686a509394aed475619c4e0b3a2df1df34fe59e90b9add8669de371eb731
@openzeppelin/contracts/access/AccessControl.sol	86908de632a9fbffc04a94fa27bd320c304a47072a85de02293e08f1724934fb
@openzeppelin/contracts/access/IAccessControl.sol	d03c1257f2094da6c86efa7aa09c1c07ebd33dd31046480c5097bc2542140e45
@openzeppelin/contracts/access/Ownable.sol	9353af89436556f7ba8abb3f37a6677249aa4df6024fbfaa94f79ab2f44f3231
@openzeppelin/contracts/security/Pausable.sol	2072248d2f79e661c149fd6a6593a8a3f038466557c9b75e50e0b001bcb5cf97
@openzeppelin/contracts/security/ReentrancyGuar	3b30604df38d0f9b2b281a3e6661eb1b9cd577579e66225c674df21ca5b89b2c

d.sol	
@openzeppelin/contracts/token/ERC20/extensions/draft-IERC20Permit.sol	3e7aa0e0f69eec8f097ad664d525e7b3f0a3fda8dcdd97de5433ddb131db86ef
@openzeppelin/contracts/token/ERC20/IERC20.sol	94f23e4af51a18c2269b355b8c7cf4db8003d075c9c541019eb8dcf4122864d5
@openzeppelin/contracts/token/ERC20/utils/SafeERC20.sol	0c8a43f12ac2081c6194d54da96f02ebc457760d6514f6b940689719fcef8c0a
@openzeppelin/contracts/utils/Address.sol	8160a4242e8a7d487d940814e5279d934e81f0436689132a4e73394bab084a6d
@openzeppelin/contracts/utils/Context.sol	1458c260d010a08e4c20a4a517882259a23a4baa0b5bd9add9fb6d6a1549814a
@openzeppelin/contracts/utils/introspection/ERC165.sol	8806a632d7b656cadb8133ff8f2acae4405b3a64d8709d93b0fa6a216a8a6154
@openzeppelin/contracts/utils/introspection/IERC165.sol	701e025d13ec6be09ae892eb029cd83b3064325801d73654847a5fb11c58b1e5
@openzeppelin/contracts/utils/math/Math.sol	8059d642ec219d0b9b62fbc76912079529cf494cac988abe5e371f1168b29b0f
@openzeppelin/contracts/utils/math/SafeMath.sol	0dc33698a1661b22981abad8e5c6f5ebca0dfe5ec14916369a2935d888ff257a

@openzeppelin/contracts/utils/Strings.sol	f81f11dca62dcd3e0895e680559676f4ba4f2e12a36bb0291d7ecbb6b983141f
contracts/crowdsale/Crowdsale.sol	75d18d26e92cbf556cfb34d575d75d035a3a181b070cd6f7fc6bf8f5b5acd332
contracts/crowdsale/distribution/FinalizableCrowdsale.sol	86b0fedc1e18aacfdfa2a1edf12c9d9d3bf32cc5868dfa50f9abd564770d5d9f
contracts/crowdsale/validation/CappedCrowdsale.sol	55f1dbe7de91970f5d3df901a284a31070ff2300f4ede6b51e35d7c2c09ebb47
contracts/crowdsale/validation/PausableCrowdsale.sol	ac8c188fe707b59659dd8a47f1b0633cc8494836570ebd3ac362d36de92b7c99
contracts/crowdsale/validation/TimedCrowdsale.sol	9bfaadf36357ac8bb9605a0181e0e93168de8bf4e99556138dd36caa3d77a9c0
contracts/crowdsale/validation/WhitelistCrowdsale.sol	cc596b4c59b93f5ff368f420df866568cab4ad05a8fe4864fe995edc89465e85
contracts/ico/PaymeTokenCrowdsale.sol	46f7170be1c8b7e721ee93c8df4c96f7a55028980c83bf81d85fa3b8a5d249c6
contracts/ico/PaymeTokenVesting.sol	89c8bf653bb3f61a0b95fad57f366f188bd72131f58878afb3482732008a1b22

Introductions

The PaymeTokenCrowdsale contract implements a crowd sale mechanism.

The users have the ability to commit/deposit Specific tokens to the crowdsale contract in exchange for a vested allocation on the crowdsale tokens. The deposited and the crowdsaled tokens will be defined once the Crowdsale contract is deployed. The vesting schedule starts on the finalization step of the crowdsale.

Roles

The owner is responsible for finalizing the crowd sale after the crowd sale has ended.

Users have the ability to participate in the crowdsale by depositing a specific type of token.

Contract Diagnostics

🔴 Critical 🟠 Medium ● Minor / Informative

Severity	Code	Description	Status
●	L02	State Variables could be Declared Constant	Unresolved
●	L09	Dead Code Elimination	Unresolved

L02 - State Variables could be Declared Constant

Criticality	minor / informative
Location	contracts/crowdsale/Crowdsale.sol#L41
Status	Unresolved

Description

Constant state variables should be declared constant to save gas.

```
_weiRaised
```

Recommendation

Add the constant attribute to state variables that never change.

L09 - Dead Code Elimination

Criticality	minor / informative
Location	contracts/crowdsale/validation/TimedCrowdsale.sol#L84,97 contracts/crowdsale/validation/WhitelistCrowdsale.sol#L23 contracts/crowdsale/Crowdsale.sol#L171,181,209 contracts/crowdsale/validation/CappedCrowdsale.sol#L46
Status	Unresolved

Description

Functions that are not used in the contract, and make the code's size bigger.

```
_preValidatePurchase
_deliverTokens
_processPurchase
_forwardFunds
_extendTime
```

Recommendation

Remove unused functions.

Contract Functions

Contract	Type	Bases		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
OwnableUpgradeable	Implementation	Initializable, ContextUpgradeable		
	__Ownable_init	Internal	✓	onlyInitializing
	__Ownable_init_unchained	Internal	✓	onlyInitializing
	owner	Public		-
	_checkOwner	Internal		
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
	_transferOwnership	Internal	✓	
Initializable	Implementation			
	_disableInitializers	Internal	✓	
	_getInitializedVersion	Internal		
	_isInitializing	Internal		
ReentrancyGuardUpgradeable	Implementation	Initializable		
	__ReentrancyGuard_init	Internal	✓	onlyInitializing
	__ReentrancyGuard_init_unchained	Internal	✓	onlyInitializing
	_nonReentrantBefore	Private	✓	
	_nonReentrantAfter	Private	✓	
IERC20PermitUpgradeable	Interface			
	permit	External	✓	-
	nonces	External		-
	DOMAIN_SEPARATOR	External		-

IERC20Upgradeable	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
SafeERC20Upgradeable	Library			
	safeTransfer	Internal	✓	
	safeTransferFrom	Internal	✓	
	safeApprove	Internal	✓	
	safeIncreaseAllowance	Internal	✓	
	safeDecreaseAllowance	Internal	✓	
	safePermit	Internal	✓	
	_callOptionalReturn	Private	✓	
AddressUpgradeable	Library			
	isContract	Internal		
	sendValue	Internal	✓	
	functionCall	Internal	✓	
	functionCall	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionStaticCall	Internal		
	functionStaticCall	Internal		
	verifyCallResultFromTarget	Internal		
	verifyCallResult	Internal		
	_revert	Private		
ContextUpgradeable	Implementation	Initializable		
	__Context_init	Internal	✓	onlyInitializing

	__Context_init_unchained	Internal	✓	onlyInitializing
	_msgSender	Internal		
	_msgData	Internal		
MathUpgradeable	Library			
	max	Internal		
	min	Internal		
	average	Internal		
	ceilDiv	Internal		
	mulDiv	Internal		
	mulDiv	Internal		
	sqrt	Internal		
	sqrt	Internal		
	log2	Internal		
	log2	Internal		
	log10	Internal		
	log10	Internal		
	log256	Internal		
	log256	Internal		
SafeMathUpgradeable	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		

AccessControl	Implementation	Context, IAccessControl, ERC165		
	supportsInterface	Public		-
	hasRole	Public		-
	_checkRole	Internal		
	_checkRole	Internal		
	getRoleAdmin	Public		-
	grantRole	Public	✓	onlyRole
	revokeRole	Public	✓	onlyRole
	renounceRole	Public	✓	-
	_setupRole	Internal	✓	
	_setRoleAdmin	Internal	✓	
	_grantRole	Internal	✓	
	_revokeRole	Internal	✓	
IAccessControl	Interface			
	hasRole	External		-
	getRoleAdmin	External		-
	grantRole	External	✓	-
	revokeRole	External	✓	-
	renounceRole	External	✓	-
Ownable	Implementation	Context		
	<Constructor>	Public	✓	-
	owner	Public		-
	_checkOwner	Internal		
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
	_transferOwnership	Internal	✓	
Pausable	Implementation	Context		
	<Constructor>	Public	✓	-

	paused	Public		-
	_requireNotPaused	Internal		
	_requirePaused	Internal		
	_pause	Internal	✓	whenNotPaused
	_unpause	Internal	✓	whenPaused
ReentrancyGuard	Implementation			
	<Constructor>	Public	✓	-
	_nonReentrantBefore	Private	✓	
	_nonReentrantAfter	Private	✓	
IERC20Permit	Interface			
	permit	External	✓	-
	nonces	External		-
	DOMAIN_SEPARATOR	External		-
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
SafeERC20	Library			
	safeTransfer	Internal	✓	
	safeTransferFrom	Internal	✓	
	safeApprove	Internal	✓	
	safeIncreaseAllowance	Internal	✓	
	safeDecreaseAllowance	Internal	✓	
	safePermit	Internal	✓	
	_callOptionalReturn	Private	✓	

Address	Library			
	isContract	Internal		
	sendValue	Internal	✓	
	functionCall	Internal	✓	
	functionCall	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionStaticCall	Internal		
	functionStaticCall	Internal		
	functionDelegateCall	Internal	✓	
	functionDelegateCall	Internal	✓	
	verifyCallResultFromTarget	Internal		
	verifyCallResult	Internal		
	_revert	Private		
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
ERC165	Implementation	IERC165		
	supportsInterface	Public		-
IERC165	Interface			
	supportsInterface	External		-
Math	Library			
	max	Internal		
	min	Internal		
	average	Internal		
	ceilDiv	Internal		
	mulDiv	Internal		
	mulDiv	Internal		
	sqrt	Internal		
	sqrt	Internal		
	log2	Internal		

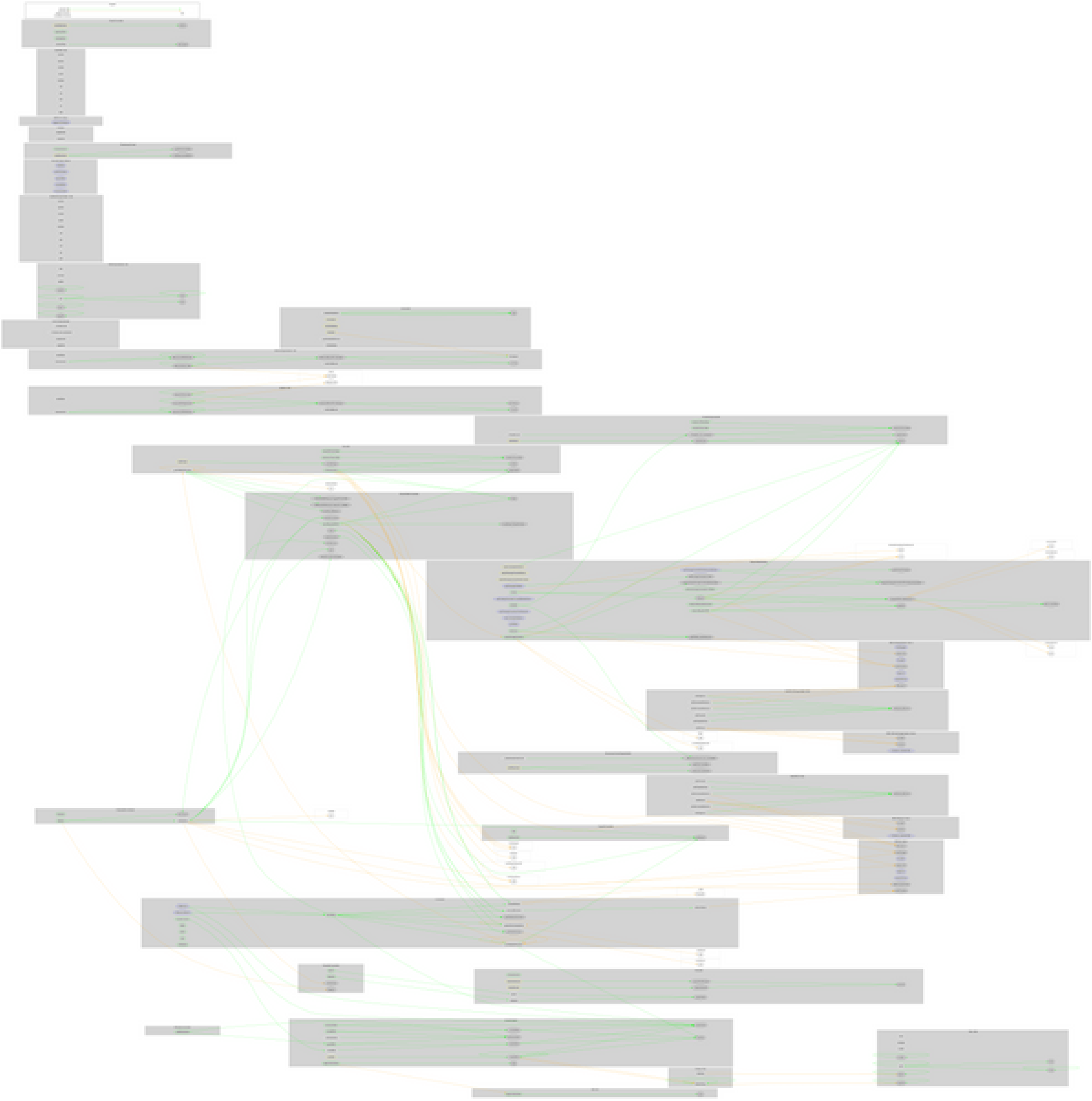
	log2	Internal		
	log10	Internal		
	log10	Internal		
	log256	Internal		
	log256	Internal		
SafeMath	Library			
	tryAdd	Internal		
	trySub	Internal		
	tryMul	Internal		
	tryDiv	Internal		
	tryMod	Internal		
	add	Internal		
	sub	Internal		
	mul	Internal		
	div	Internal		
	mod	Internal		
	sub	Internal		
	div	Internal		
	mod	Internal		
Strings	Library			
	toString	Internal		
	toHexString	Internal		
	toHexString	Internal		
	toHexString	Internal		
Crowdsale	Implementation	Context, ReentrancyGuard, AccessControl		
	<Constructor>	Public	✓	-
	<Fallback>	External	Payable	-
	<Receive Ether>	External	Payable	-
	token	Public		-
	wallet	Public		-

	rate	Public		-
	weiRaised	Public		-
	buyTokens	Public	Payable	nonReentrant
	_preValidatePurchase	Internal		
	_postValidatePurchase	Internal		
	_deliverTokens	Internal	✓	
	_processPurchase	Internal	✓	
	_updatePurchasingState	Internal	✓	
	_getTokenAmount	Internal		
	_forwardFunds	Internal	✓	
FinalizableCrowdsale	Implementation	TimedCrowdsale		
	<Constructor>	Public	✓	-
	finalized	Public		-
	finalize	Public	✓	-
	_finalization	Internal	✓	
CappedCrowdsale	Implementation	Crowdsale		
	<Constructor>	Public	✓	-
	cap	Public		-
	capReached	Public		-
	_preValidatePurchase	Internal		
PausableCrowdsale	Implementation	Crowdsale, Pausable, Ownable		
	_preValidatePurchase	Internal		whenNotPaused
	pause	Public	✓	onlyOwner whenNotPaused
	unpause	Public	✓	onlyOwner whenPaused
TimedCrowdsale	Implementation	Crowdsale		

	<Constructor>	Public	✓	-
	openingTime	Public		-
	closingTime	Public		-
	isOpen	Public		-
	hasClosed	Public		-
	_preValidatePurchase	Internal		onlyWhileOpen
	_extendTime	Internal	✓	
WhitelistCrowdsale	Implementation	AccessControl, Crowdsale		
	_preValidatePurchase	Internal		
	addWhitelisted	Public	✓	onlyRole
PaymeTokenCrowdsale	Implementation	Ownable, CappedCrowdsale, TimedCrowdsale, WhitelistCrowdsale, FinalizableCrowdsale, PausableCrowdsale		
	<Constructor>	Public	✓	Crowdsale CappedCrowdsale TimedCrowdsale
	buyTokensInBUSD	Public	Payable	nonReentrant
	buyTokens	Public	Payable	nonReentrant
	_forwardFunds	Internal	✓	
	_preValidatePurchase	Internal		
	createInvestor	Internal	✓	
	_processPurchase	Internal	✓	
	_updatePurchasingState	Internal	✓	
	_finalization	Internal	✓	
	createInvestors	Public	✓	-
	finalize	Public	✓	onlyOwner

PaymeTokenVesting	Implementation	OwnableUpgradeable, ReentrancyGuardUpgradeable		
	initialize	Public	✓	initializer
	getVestingSchedulesCountByBeneficiary	External		-
	getVestingIdAtIndex	External		-
	getVestingScheduleByAddressAndIndex	External		-
	getVestingSchedulesTotalAmount	External		-
	setCrowdsaleAddress	External	✓	-
	getToken	External		-
	createVestingSchedule	Public	✓	onlyCrowdsaleOrOwner
	revoke	Public	✓	onlyOwner onlyIfVestingScheduleNotRevoked
	withdraw	Public	✓	nonReentrant onlyOwner
	releaseTokenForTGE	Public	✓	nonReentrant
	release	Public	✓	nonReentrant onlyIfVestingScheduleNotRevoked
	getVestingSchedulesCount	Public		-
	computeReleasableAmount	Public		onlyIfVestingScheduleNotRevoked
	getVestingSchedule	Public		-
	getWithdrawableAmount	Public		-
	computeNextVestingScheduleIdForHolder	Public		-
	getLastVestingScheduleForHolder	Public		-
	computeVestingScheduleIdForAddressAndIndex	Public		-
	_computeReleasableAmount	Internal		
	getCurrentTime	Public		-

Contract Flow



Domain Info

Domain Name	payme.games
Registry Domain ID	29f4ee9286e043058b41ccc27375747f-DONUTS
Creation Date	2021-01-06T13:00:37Z
Updated Date	2022-08-05T11:31:27Z
Registry Expiry Date	2023-01-06T13:00:37Z
Registrar WHOIS Server	whois.namecheap.com
Registrar URL	https://www.namecheap.com/
Registrar	NameCheap, Inc.
Registrar IANA ID	1068

The domain was created almost 2 years before the creation of the audit. It will expire in 29 days.

There is no public billing information, the creator is protected by the privacy settings.

Summary

The PaymeTokenCrowdsale contract is responsible for exchanging BUSD for native tokens in order to vest them. This audit investigates security issues and mentions business logic concerns and potential improvements.

We state that owner privileges are necessary and required for proper protocol operations. Thus, we emphasize the contract owner be extra careful with the credentials.

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