Cyberscope

Audit Report

SquadSwap

November 2023

Repository	https://github.com/Bit5Tech
Projects	SquadSwap, SquadToken, SquadSwap-v3
Audited by	© cyberscope

Table of Contents

Review

Repository	https://github.com/Bit5Tech
SquadSwap Commit	0a65867700e63d4be7083c7911f4da06854d0337
SquadToken Commit	f3718817028eb1b862b153b9a36e3307e9271d60
SquadSwap-v3 Commit	4b84122cb1116545be78851ae263dc11a63a499c

Audit Updates

Initial Audit	24 Nov 2023 https://github.com/cyberscope-io/audits/blob/main/squadswap/v1/audit.pdf
Corrected Phase 2	28 Nov 2023

Source Files

Filename	SHA256
/Squad.sol	e055cd4ec165842b536818ba2984626cdd8d42e08cd5d73a984b0faf03220433
/v3/V3Migrator.sol	b08d9a41f2d9c01e994c2fa292b7cbcdcb33b2d254c10d8ae4cc068eaa2a9fcd
/v3/UnsafeMath.sol	4d02353eb503e3111e25bd50104ac9b279f99e88d848e455262a3fbeb55c50e7
/v3/TransferHelper.sol	ccb87429b290eb6ed429648a7131f68ce0151a74f3ed27de78aacd28015e4590
/v3/TickMath.sol	a2aad5dd689b688481ccc73e70416ff36abbc20f0bee15e2ff22336b492dca83
/v3/TickBitmap.sol	8452c484e6caad95411358d8c1763810e715b3d13697c83665657619472d3b0a
/v3/Tick.sol	d938c31db4532ea087d90c38379ddc0a4ee5709b44a421abf87961e0730d008c
/v3/SwapRouter.sol	6ca6c830990aea47fed8b728e87c6161d2bc463e0292b7ea2ea9f080e25c4183
/v3/SwapMath.sol	cdb205f8790e6c8a3587bd3db6eec6fba874afca1c0c6e890d87452f7aadc902
/v3/SquadV3PoolDeployer.sol	ecd80fb2a07fcac1e2a79f5ed3c4349ddfcf341948f0dd09cd15e3c70e2e3740
/v3/SquadV3Pool.sol	154e13fc77c65722ea732896df044c79d04c40f2873e7de2bc1605b74e8a7d38
/v3/SquadV3Factory.sol	6bfe0bc1962aa5eb465f9f882552d9cfaa85cf438aa9efcb529f14737fbaa192
/v3/SqrtPriceMath.sol	36eeb343e0b1809cd76b2ec72a336923aa24f857965966543b065e660b2ebc6e

/v3/SelfPermit.sol	48bb499a5e2bb8063788faf42ba0abd71cbd63392aa4d4c12531b530419d6afa
/v3/SafeCast.sol	9aed494b56d3dd16b7d6535583ded2cdfb03dc80aaa919347b13d35fd597e8bf
/v3/Position.sol	d87d5ecd8531d9311e0953462b56ccd6453b65107cc62f43602f59a4edccb806
/v3/PoolAddress.sol	178a6134c6e7572b96df782d334d834dca3027330aad4b0da29ff17c25fff141
/v3/PeripheryPaymentsWithFee.sol	6ef994f772d796f196aa7491c684b443c72a5dee223a3f04b0afbbbe82319c65
/v3/PeripheryPayments.sol	68cef83e01906a13f4a2bb1c12a9e99fad3e957eea6ddbb54bac30ba3b06a436
/v3/PeripheryImmutableState.sol	f4611f54f13d0599648bf88fc5bba7fe8eb3bfc27f898c5cc0e2f27272ebca99
/v3/Path.sol	42edaa8b6c577bee7a24b2f1d377fa7fb7649526a935040ccdd1a91a7f3b46a0
/v3/Oracle.sol	e77c590445158e991b377da4ce33d42c98d5ac842cdd1ad6cf1c7ba4c541a457
/v3/NonfungibleTokenPositionDescriptorOffChainV2.sol	9041d1e442dd614203d15079de17c3a2de449930b65f4a732db0bf1893f382ff
/v3/NonfungibleTokenPositionDescriptor.sol	2a8f4a8eb154c9c8422f479b41c760c3d9bb05e99b9c6b10d4497f92e58d4a59
/v3/NonfungiblePositionManager.sol	ca9f4b4c105e10c86a15ed14e0598d057580282a82840a191937d3fd9ba47622
/v3/NFTDescriptorEx.sol	40ff1c73bf6ee2160ef6a9801711daffc1cc33e0315ae9f8afd0b8ec6957efe8
/v3/Multicall.sol	029ad0bcade48ff32da51094a3fb245fd7d8324c4fb4dd20fb4b2614efc9618c

/v3/MasterChefV3.sol	120678ea48e82cf26a15b78885607a9e683d59b110f54c996c9bb606c0a7aecf
/v3/LowGasSafeMath.sol	394107ff2dbbaded5612452af5e77b4af9d0871b096c1514b0ea659b862fc46f
/v3/LiquidityMath.sol	84d20a16d5346f6ec4c12dff4df23dda5d46e52d33f18aaaaac2e9e36ce4a072
/v3/IWETH9.sol	e0394b612c06c730766ab904f7a2a090e2b37ebeee4dd5e922d7213342c6a519
/v3/IWETH.sol	175ac0e2656842d563d6a1a3a6ade75350d8bc65af4c0e81626d05de6f182568
/v3/ISwapRouter.sol	bbe3b1fb1acd3801ba2e20fcf60f488c9158992b7ee25b060ffdddc5a22db653
/v3/ISquadV3SwapCallback.sol	1bce1224b428455c1c519ec2a0f2a305457d26fd8335a0a4551fe5e7f754275d
/v3/ISquadV3PoolOwnerActions.sol	91326209c643d4195e7d6c9ed1608f14ffaab0fb9dd0e2228360812822c0a10e
/v3/ISquadV3PoolImmutables.sol	1a85012d37bf56b28b3aa9141481b4e6c33eeb1583a7d1ed663c2cef1fa6dc80
/v3/ISquadV3PoolImmutables (1).sol	1a85012d37bf56b28b3aa9141481b4e6c33eeb1583a7d1ed663c2cef1fa6dc80
/v3/ISquadV3PoolEvents.sol	a28e85dfef396593743631665a89ba0907fe46652cb0ecc2b2f9e00640538080
/v3/ISquadV3PoolDeployer.sol	98d49665bc6fd783251cedd883898de98b77bf4f6cf7a3709a8080fadd8e07b6
/v3/ISquadV3PoolActions.sol	64df04567e572e9e439abd086fea7e6407c9e0a37385497e3b1df9f137980828
/v3/ISquadV3Pool.sol	8f4429dbf79387b6b65c8a450039f628620523531f1a3929373445a8c8b8a815

/v3/ISquadV3MintCallback.sol	c9f43d615d4972c88f4f45f4614a3f0b24b544516ca3d4435c1d253795d05db4
/v3/ISquadV3LmPool.sol	6c805e3b27a557e157df823712c7a0de13961fa0a565c5f7dceb89c878a6a8bc
/v3/ISquadV3FlashCallback.sol	01f167a0380edec605e2377a0381dec748472952987616e19358e51aa162e536
/v3/ISquadV3Factory.sol	60cf8b5b55e972768f8991d4d2ed6dce093bd7a00bf829d2d4aebef8f707aebe
/v3/IPeripheryPayments.sol	5cee8018aba8e8b59d0704360cfd07b28a06b6d84c4e2c61ae1fedf3a3e406a0
/v3/IPeripheryImmutableState.sol	09a6b62a34b824cd6e57e6a1ac0de8514bcc8cd983c3a09fca1cf263aeb4142b
/v3/INonfungiblePositionManagerStruct.sol	b5e107cc467ad2b47aa417ffbaef8659c613cb413a428d6c8ed442e4ac14018f
/v3/INonfungiblePositionManager.sol	dfe01177b3410e3bde361c1ef16fed65bd584322c6b9ba8f4a97a63009a80d40
/v3/IMasterChefV2.sol	94ec05df38a733410f7b2291048604b40271f5276442a1ee39a07921247ae96a
/v3/ILMPoolDeployer.sol	fbbca9e76f52c39c024481bbe1a627d4408fd8f3eae21986a70266d590ba6bcf
/v3/ILMPool.sol	d6d2deb9b01ca89c110b9f802d42b9fe377284a27018e626a7153837c0da7fc7
/v3/IFarmBooster.sol	fb99be458b9adfe9c3b51ff5e0bf77795837b2acc8e327244c6d3ff62ceedb52
/v3/IERC20PermitAllowed.sol	97a8d607694194b6de26c0a15d9439394748e8157fa38813bc7614f7d2e4ba67
/v3/IERC20Minimal.sol	d3de1555cfb2fab915842aa30fea44f6429386ed01be5256ac2e260b5944fabd

/v3/FullMath.sol	0a18f00afc2b99b3226898303319bf0a9108ace44c8871491571f53de2f0bf0d
/v3/FixedPoint96.sol	219deb88ffbcdefa482be35051db586378e8523062bee592dd2c5fa7fb47ebd6
/v3/FixedPoint128.sol	cfc3aef8851f183492547dccc168bf72398fba2aad4c4d9d4784f542a8ccda34
/v3/FeeManager.sol	f08689c2474cc5e7629eeae0ce1d33b4e1e37d8233be0a4dd0b280aac6d9bce7
/v3/Enumerable.sol	b07a199e4befd5186d6e5d6307ffba3b09b1ae8e6b78549ae41dc37a8c714aca
/v3/CallbackValidation.sol	c13dc106b8c87c4474dc5ab22ddb74f3c927fbaf751dab1d63218672463fd199
/v3/BytesLib.sol	abe5da07d5e9f890fc64ca7b9283fa88a81a0909e4510452bdfb470d4d49bddf
/v3/BlockTimestamp.sol	e5ca9a8b6b9e0cafcd9a9966b05228a1572f82fccee396d2e0eff5f8aa9bb1f4
/v3/BitMath.sol	32f71ea9156f55572a72efb0b2a913df88de66ff33d042043fb3e51a6050a557
/v2/UQ112x112.sol	915f7e58b9971ea3a5f869880e5c6f434066d30c561117429e60eafd9ddc2fcd
/v2/SquadswapRouter02.sol	244590415d1f512065642559327b86d5757bbe99ac81491ab46123fce8f5980c
/v2/SquadswapPair.sol	66bf09e7207d400b6e446b0970a52f3b27498d084521ab29cfa55dcb7397cc77
/v2/SquadswapLibrary.sol	d900424e8a98e5fb1101d3f49b4b1a7d07b15c58cf50284ced6531b870e76229
/v2/SquadswapFactory.sol	63f8c1588c649819474952ee4782730ebbdaec3f9884e011ed128c0949c2d8de

/v2/SquadswapERC20Test.sol	c77f25472ccb79abc63a29ed7bd182ddcb994104a384edf41502b5e7c27f2226
/v2/SquadswapERC20.sol	61b908de89647ecd3a22322c96f2dc5ddccad86a183057154d095cca14ece842
/v2/MathTest.sol	580b0fbbaad62467d23b12731d477a2b8eaa8d6e4c3c3319969ba035df0d1c65
/v2/Math.sol	ed421bb65a57c163f8f9f121387142948ba9bea65ae4f80601902eeb14a5d755
/v2/IWETH.sol	9480494adfb02acd3c5c6850655c8b44c7bd21b95e0f6c89e2fc6cbb6c046806
/v2/IUniswapV1Factory.sol	40cdb3ba46070597ddb17db11c1223e4a0850fc38f9dc1f8c82d86dc602379c9
/v2/IUniswapV1Exchange.sol	28f6a8777b2c8ec5ca5113554740a24464ec8678de3f380e566497f00c1081af
/v2/ISquadswapRouter01.sol	64ffcb3f3d0132505ee2b05d360b7d9b15b2b14807c7b87b9ac1d9a7f2d88774
/v2/ISquadswapPair.sol	c31ecaed4c02f25a2fb39060748c332631e828544825283988cbf27a7c148073
/v2/ISquadswapMigrator.sol	72bce8b652c5807c1e1f27b89c922c506c0a0118e79009f4fb98af189dcddf53
/v2/ISquadswapFactory.sol	e05d9c3e844f8facb673c118b59fccdc42da3d9a06a40c5758852a2f308d8239
/v2/ISquadswapERC20.sol	b065daeda7cf4f8c8211ff0369b4eca2d71e34c11db4fa9ddad58edaa9c9d2bb
/v2/ISquadswapCallee.sol	ef8a9325a374498ea5e1c9451d7b703034502027f99bd083c331b86384cbc431
/v2/IERC20.sol	732545ddf5ddf609ac15b58ae52e9319bd03235143ca3599e6f08be6fdc782fa

/v2/IBEP20.sol	36213f63b573b18220550eebd3a17e42aa0658caf53c24353478bdce5fc809a2
/v2/DeflatingERC20.sol	387e2a62f4b6b2ff2cd34ba43f096ab01ad742793540987dd558e7ff41bc01cb

Overview

SquadToken

The `SquadToken` contract, is an implementation of a BEP20 token on the Binance Smart Chain (BSC). This contract adheres to the standards set by the BEP20 protocol, ensuring compatibility and functionality within the BSC ecosystem. The contract is structured to provide a secure and efficient means of creating and managing digital assets, leveraging the robustness of blockchain technology. It includes essential features such as token transfer, balance queries, and allowance management, which are fundamental to the operation of any digital token on a blockchain network.

At the core of the SquadToken contract is the implementation of key functionalities that define its behavior and utility. The contract includes mechanisms for ownership management, allowing the initial deployer of the contract to be designated as the owner. This ownership can be transferred or renounced, providing flexibility and control over the contract's administration. Additionally, the contract incorporates the SafeMath library, a critical component for ensuring safe arithmetic operations, thereby mitigating risks such as overflow and underflow errors. This inclusion is particularly important in the context of financial transactions and token management, where accuracy and security are paramount.

Furthermore, the SquadToken contract is designed with user-centric features that enhance its usability within the BSC network. It supports standard BEP20 functions like transferring tokens between accounts, approving third parties to spend tokens on behalf of the token holder, and querying token balances. These functions are integral to the token's interaction with other contracts and users on the network. The contract's adherence to the BEP20 standard ensures that it can seamlessly integrate with a wide range of decentralized applications (dApps) and services within the Binance Smart Chain ecosystem, making it a versatile and valuable asset for various blockchain-based applications.

V2 Contracts

The SquadSwap contracts, stand as a decentralized finance (DeFi) application on the blockchain. They are designed for automated token exchange, leveraging a suite of smart contracts, each renamed to align with the SquadSwap theme. The core contracts include `SquadswapERC20.sol` , which serves as a template for ERC20 tokens within the ecosystem, and `SquadswapFactory.sol` , a central component for creating new liquidity pairs for any two tokens. Additionally, the `SquadswapPair.sol` contract represents individual liquidity pools for token pairs, handling key functionalities like liquidity provision and token swaps.

The platform's architecture is further supported by a range of interfaces and utility contracts. These include `IERC20.sol` and `IBEP20.sol` , ensuring compatibility with a wide range of tokens. The `ISquadswapCallee.sol` interface allows for the implementation of custom logic in response to token transfers. The platform also includes contracts like `SquadswapRouter02.sol` for facilitating multi-step transactions.

Moreover, SquadSwap incorporates various utility and safety contracts such as `Math.sol`, `SafeMath.sol` , and `UQ112x112.sol` , crucial for precise financial calculations and preventing common vulnerabilities. The ecosystem is rounded off with testing and validation contracts like `DeflatingERC20.sol` and `SquadswapERC20Test.sol` , ensuring the platform's functionality and security.

Overall, SquadSwap offers a comprehensive DeFi solution, enabling seamless token swaps, liquidity provision, and a range of other financial activities in a decentralized and secure environment.

V3 Contracts

SquadSwap's adaptation of the V3 protocol introduces a sophisticated and flexible DeFi platform on the blockchain, tailored to offer enhanced liquidity and trading features.

At the base of this ecosystem is the `SquadV3Factory.sol`, a pivotal contract responsible for creating and managing liquidity pools, known as SquadV3Pools. These pools, defined in `SquadV3Pool.sol` and `SquadV3PoolDeployer.sol`, are where liquidity providers can add their assets and traders can swap tokens. The unique feature of these pools is their concentrated liquidity, allowing liquidity providers to allocate their capital within specific price ranges, optimizing capital efficiency.

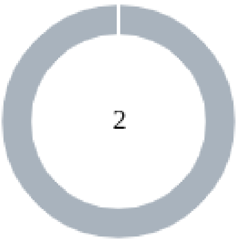
The `NonfungiblePositionManager.sol` and `NonfungibleTokenPositionDescriptor.sol` contracts play a crucial role in managing liquidity positions, which are represented as NFTs. This approach allows for more granular control and flexibility over individual liquidity positions. The `SwapRouter.sol` and `ISwapRouter.sol` contracts facilitate the actual token swaps, offering an interface for users to easily trade tokens within the SquadSwap ecosystem.

Safety and utility are also paramount in this ecosystem. Contracts like `SafeCast.sol`, `TickMath.sol`, and `LowGasSafeMath.sol` ensure accurate and efficient mathematical operations, crucial for financial transactions. The `FeeManager.sol` and `PeripheryPaymentsWithFee.sol` contracts handle the platform's fee structure and distribution, ensuring a fair and sustainable system.

Furthermore, the platform includes advanced features like the `V3Migrator.sol` for migrating liquidity from previous versions, and the `MasterChefV3.sol` for incentivizing liquidity provision. The integration of `IWETH.sol` and `IWETH9.sol` ensures seamless interaction with the wrapped token, a key aspect of operating on the blockchain.

In summary, SquadSwap's implementation of the UniswapV3 protocol on the blockchain offers a robust and feature-rich DeFi platform. It provides users with advanced trading and liquidity provision options, enhanced by the concentrated liquidity feature, while ensuring security and efficiency through a suite of carefully designed smart contracts.

Findings Breakdown



●	Critical	0
●	Medium	0
●	Minor / Informative	2

Severity		Unresolved	Acknowledged	Resolved	Other
●	Critical	0	0	0	0
●	Medium	0	0	0	0
●	Minor / Informative	2	0	0	0

Diagnostics

● Critical ● Medium ● Minor / Informative

Severity	Code	Description	Status
●	L16	Validate Variable Setters	Unresolved
●	L19	Stable Compiler Version	Unresolved

L16 - Validate Variable Setters

Criticality	Minor / Informative
Location	v2/factory/SquadswapPair.sol#L67,68 v2/factory/SquadswapFactory.sol#L19,45,50
Status	Unresolved

Description

The contract performs operations on variables that have been configured on user-supplied input. These variables are missing of proper check for the case where a value is zero. This can lead to problems when the contract is executed, as certain actions may not be properly handled when the value is zero.

```
token0 = _token0
token1 = _token1
feeToSetter = _feeToSetter
feeTo = _feeTo
```

Recommendation

By adding the proper check, the contract will not allow the variables to be configured with zero value. This will ensure that the contract can handle all possible input values and avoid unexpected behavior or errors. Hence, it can help to prevent the contract from being exploited or operating unexpectedly.

L19 - Stable Compiler Version

Criticality	Minor / Informative
Location	v2/factory/SquadswapPair.sol#L2 v2/factory/SquadswapFactory.sol#L2 v2/factory/SquadswapERC20.sol#L2 v2/factory/libraries/UQ112x112.sol#L2 v2/factory/libraries/Math.sol#L2 v2/factory/interfaces/ISquadswapPair.sol#L2 v2/factory/interfaces/ISquadswapFactory.sol#L2 v2/factory/interfaces/ISquadswapCallee.sol#L2 v2/factory/interfaces/IERC20.sol#L2
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.

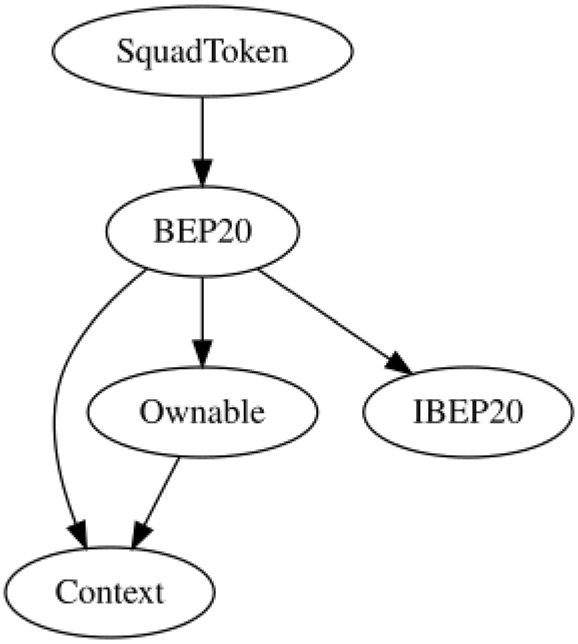
Functions Analysis

Contract	Type	Bases		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
Context	Implementation			
	_msgSender	Internal		
Ownable	Implementation	Context		
		Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
	_transferOwnership	Internal	✓	
IBEP20	Interface			
	totalSupply	External		-
	decimals	External		-
	symbol	External		-
	name	External		-
	getOwner	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-

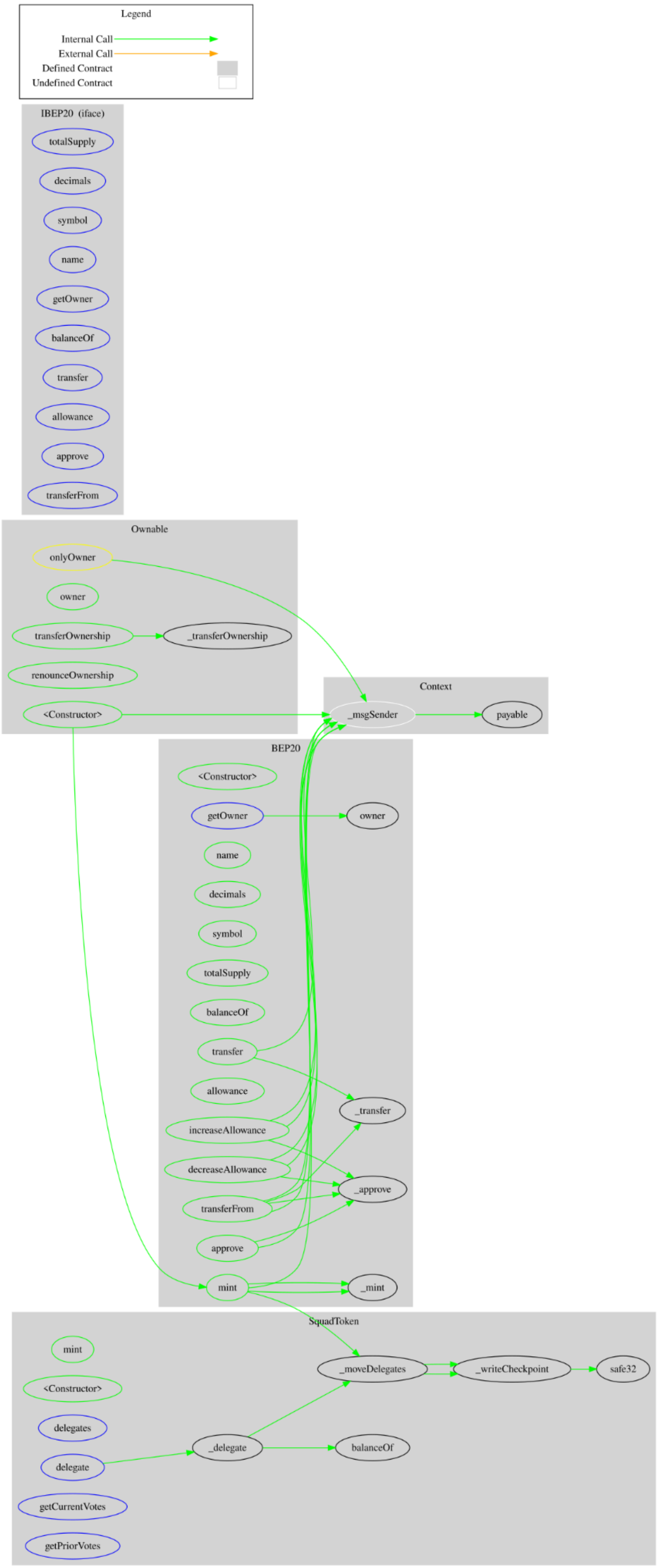
	approve	External	✓	-
	transferFrom	External	✓	-
BEP20	Implementation	Context, IBEP20, Ownable		
		Public	✓	-
	getOwner	External		-
	name	Public		-
	decimals	Public		-
	symbol	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-
	transferFrom	Public	✓	-
	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	mint	Public	✓	onlyOwner
	_transfer	Internal	✓	
	_mint	Internal	✓	
	_approve	Internal	✓	
SquadToken	Implementation	BEP20		

	mint	Public	✓	onlyOwner
		Public	✓	-
	delegates	External		-
	delegate	External	✓	-
	getCurrentVotes	External		-
	getPriorVotes	External		-
	_delegate	Internal	✓	
	_moveDelegates	Internal	✓	
	_writeCheckpoint	Internal	✓	
	safe32	Internal		

Inheritance Graph



Flow Graph


Legend
Internal Call
External Call
Defined Contract
Undefined Contract
IBEP20 (iface)
totalSupply
decimals
symbol
name
getOwner
balanceOf
transfer
allowance
approve
transferFrom
Ownable
onlyOwner
owner
transferOwnership
_transferOwnership
renounceOwnership
<Constructor>
Context
_msgSender
payable
BEP20
<Constructor>
getOwner
owner
name
decimals
symbol
totalSupply
balanceOf
transfer
allowance
_transfer
increaseAllowance
decreaseAllowance
_approve
transferFrom
approve
mint
_mint
SquadToken
mint
<Constructor>
_moveDelegates
_writeCheckpoint
safe32
delegates
delegate
_delegate
balanceOf
getCurrentVotes
getPriorVotes

Summary

SquadSwap contract implements a token mechanism. This audit investigates security issues, business logic concerns and potential improvements.

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