



# Smart Contract Audit

FOR

Squishy

DATED : 16 Sep 24'



# MANUAL TESTING

---

**Centralization** – The owner can Blacklist Wallet.

**Severity: High**

**Function: blacklist**

**Status: Open**

**Overview:**

The owner can blacklist multiple wallets.

```
function blacklist(address _address, bool _isBlacklisting) external onlyOwner {  
    blacklists[_address] = _isBlacklisting;  
}
```

**Suggestion:**

There should be a locking period so that the wallet cannot be locked in an

indefinite

Period of time.



# AUDIT SUMMARY

**Project name - Squishy**

**Date:** 16 Sep, 2024

**Scope of Audit-** Audit Ace was consulted to conduct the smart contract audit of the solidity source codes.

**Audit Status: High Risk Major Flag**

## Issues Found

Status	Critical	High	Medium	Low	Suggestion
Open	0	1	0	2	1
Acknowledged	0	0	0	0	0
Resolved	0	0	0	0	0



# USED TOOLS

---

## Tools:

### 1- Manual Review:

A line by line code review has been performed by audit ace team.

**2- BSC Test Network:** All tests were conducted on the BSC Test network, and each test has a corresponding transaction attached to it. These tests can be found in the "Functional Tests" section of the report.

### 3- Slither :

The code has undergone static analysis using Slither.

### Testnet version:

The tests were performed using the contract deployed on the BSC Testnet, which can be found at the following address:

<https://testnet.bscscan.com/address/0xe6dc4cc1cf81cf832821bed437005d865555863c#code>

---



# Token Information

---

**Token Address:**

0x87b5c6F208cd08AE0F84bEa7672Cf23b937948E1

**Name:** Squishy

**Symbol:** SQUISHY

**Decimals:** 18

**Network:** Ethereum Network

**Token Type:** ERC-20

**Owner:** 0x0e370dE3E84E3Bf21ed70e3d4DcEF4941B12E61b

**Deployer:** 0x0e370dE3E84E3Bf21ed70e3d4DcEF4941B12E61b

**Token Supply:** 1000000000

**Checksum:** Ac6659e84744e0102ab19c1d1e78a21a

**Testnet:**

<https://testnet.bscscan.com/address/0xe6dc4cc1cf81cf832821bed437005d865555863c#code>

---



# TOKEN OVERVIEW

---

**Buy Fee:** 0-0%

---

**Sell Fee:** 0-0%

---

**Transfer Fee:** 0-0%

---

**Fee Privilege:** Owner

---

**Ownership:** Owned

---

**Minting:** None

---

**Max Tx:** None

---

**Blacklist:** Yes

---





# AUDIT METHODOLOGY

---

The auditing process will follow a routine as special considerations by Auditace:

- Review of the specifications, sources, and instructions provided to Auditace to make sure the contract logic meets the intentions of the client without exposing the user's funds to risk.
- Manual review of the entire codebase by our experts, which is the process of reading source code line-by-line in an attempt to identify potential vulnerabilities.
- Specification comparison is the process of checking whether the code does what the specifications, sources, and instructions provided to Auditace describe.
- Test coverage analysis determines whether the test cases are covering the code and how much code is exercised when we run the test cases.
- Symbolic execution is analysing a program to determine what inputs cause each part of a program to execute.
- Reviewing the codebase to improve maintainability, security, and control based on the established industry and academic practices.

# VULNERABILITY CHECKLIST



Return values of low-level calls



**Gasless Send**



Private modifier



Using block.timestamp



Multiple Sends



Re-entrancy



Using Suicide



Tautology or contradiction



Gas Limit and Loops



Timestamp Dependence



Address hardcoded



Revert/require functions



Exception Disorder



Use of tx.origin



Using inline assembly



Integer overflow/underflow



Divide before multiply



Dangerous strict equalities



Missing Zero Address Validation



Using SHA3

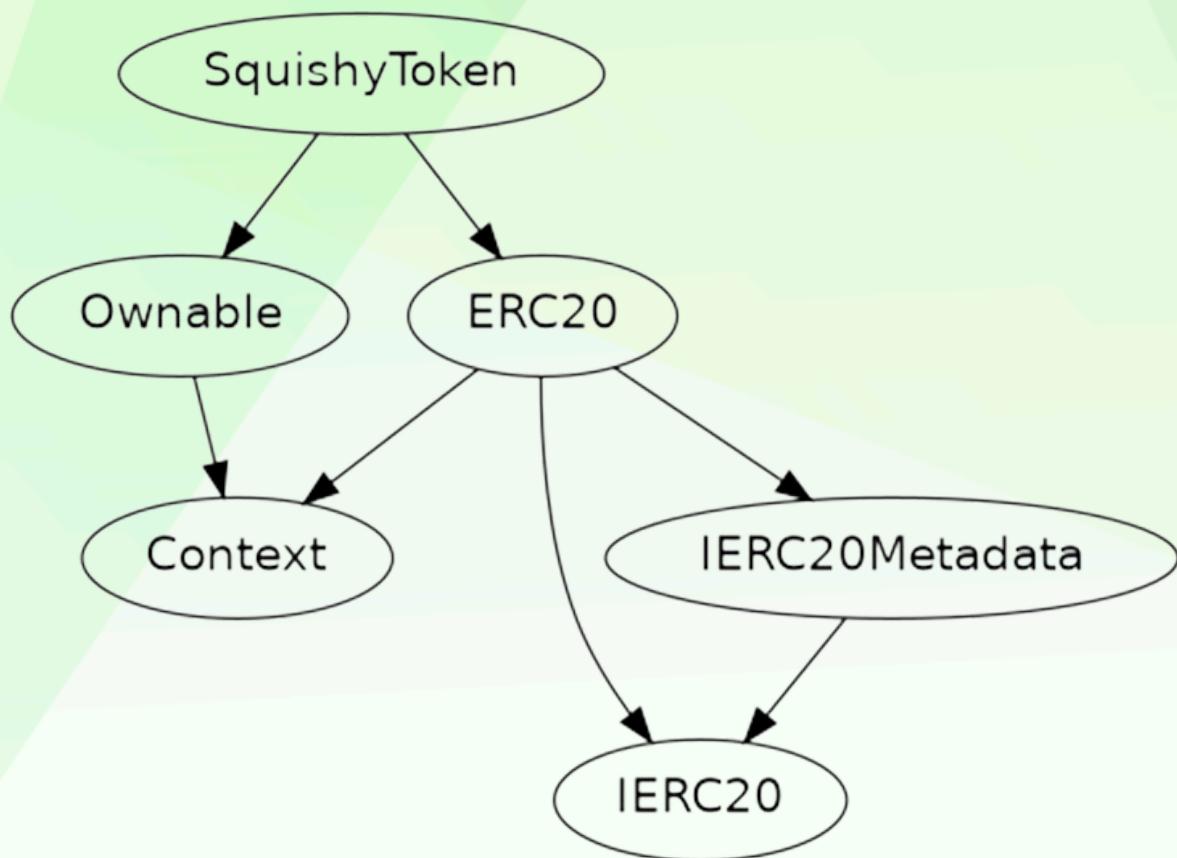


Compiler version not fixed



Using throw

# INHERITANCE TREE





## POINTS TO NOTE

---

- The owner can transfer ownership.
- The owner can renounce ownership.
- The owner can blacklist the address.
- The owner can set the rule.



# STATIC ANALYSIS

```
INFO:Detectors:  
SquishyToken.constructor(uint256)._totalSupply (SquishyToken.sol#599) shadows:  
    - ERC20._totalSupply (SquishyToken.sol#266) (state variable)  
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#local-variable-shadowing  
INFO:Detectors:  
SquishyToken.setRule(bool,address,uint256,uint256) (SquishyToken.sol#607-612) should emit an event for:  
    - maxHoldingAmount = _maxHoldingAmount (SquishyToken.sol#610)  
    - minHoldingAmount = _minHoldingAmount (SquishyToken.sol#611)  
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#missing-events-arithmetic  
INFO:Detectors:  
SquishyToken.setRule(bool,address,uint256,uint256)._uniswapV2Pair (SquishyToken.sol#607) lacks a zero-check on :  
    - uniswapV2Pair = _uniswapV2Pair (SquishyToken.sol#609)  
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#missing-zero-address-validation  
INFO:Detectors:  
2 different versions of Solidity are used:  
    - Version constraint ^0.8.0 is used by:  
        -^0.8.0 (SquishyToken.sol#10)  
        -^0.8.0 (SquishyToken.sol#38)  
        -^0.8.0 (SquishyToken.sol#116)  
        -^0.8.0 (SquishyToken.sol#202)  
        -^0.8.0 (SquishyToken.sol#232)  
    - Version constraint ^0.8.17 is used by:  
        -^0.8.17 (SquishyToken.sol#589)  
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#different-pragma-directives-are-used  
INFO:Detectors:  
Context._msgData() (SquishyToken.sol#27-29) is never used and should be removed  
ERC20._beforeTokenTransfer(address,address,uint256) (SquishyToken.sol#557-561) is never used and should be removed  
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dead-code  
INFO:Detectors:  
Version constraint ^0.8.0 contains known severe issues (https://solidity.readthedocs.io/en/latest/bugs.html)  
    - FullInlinerNonExpressionSplitArgumentEvaluationOrder  
    - MissingSideEffectsOnSelectorAccess  
    - AbiReencodingHeadOverflowWithStaticArrayCleanup  
    - DirtyBytesArrayToStorage  
    - DataLocationChangeInInternalOverride  
    - NestedCalldataArrayAbiReencodingSizeValidation  
    - SignedImmutables  
    - ABIDecodeTwoDimensionalArrayMemory  
    - KeccakCaching.  
  
INFO:Detectors:  
Parameter SquishyToken.blacklist(address,bool)._address (SquishyToken.sol#603) is not in mixedCase  
Parameter SquishyToken.blacklist(address,bool)._isBlacklisting (SquishyToken.sol#603) is not in mixedCase  
Parameter SquishyToken.setRule(bool,address,uint256,uint256)._limited (SquishyToken.sol#607) is not in mixedCase  
Parameter SquishyToken.setRule(bool,address,uint256,uint256)._uniswapV2Pair (SquishyToken.sol#607) is not in mixedCase  
Parameter SquishyToken.setRule(bool,address,uint256,uint256)._maxHoldingAmount (SquishyToken.sol#607) is not in mixedCase  
Parameter SquishyToken.setRule(bool,address,uint256,uint256)._minHoldingAmount (SquishyToken.sol#607) is not in mixedCase  
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-solidity-naming-conventions  
INFO:Slither:SquishyToken.sol analyzed (6 contracts with 94 detectors), 14 result(s) found
```

**Result => A static analysis of contract's source code has been performed using slither, No major issues were found in the output**



# FUNCTIONAL TESTING

---

## 1- Approve (passed):

<https://testnet.bscscan.com/tx/0x7ac7409a3bf4d9e1e0e560a858c89f19ede5a95f4ec6bc335993360151d30982>

## 2- Blacklist (passed):

<https://testnet.bscscan.com/tx/0x9a44ddd58b1a91c040d5dfd74e7f3ed611dd8172da012bbab008b2be2544499b>

## 3- Set Rule (passed):

<https://testnet.bscscan.com/tx/0xefd9b467cdb1af79e434ee699df0d685f3fd96ca6adad4002eff5153a292d92f>



# CLASSIFICATION OF RISK

Severity	Description
◆ Critical	These vulnerabilities could be exploited easily and can lead to asset loss, data loss, asset, or data manipulation. They should be fixed right away.
◆ High-Risk	A vulnerability that affects the desired outcome when using a contract, or provides the opportunity to use a contract in an unintended way.
◆ Medium-Risk	A vulnerability that could affect the desired outcome of executing the contract in a specific scenario.
◆ Low-Risk	A vulnerability that does not have a significant impact on possible scenarios for the use of the contract and is probably subjective.
◆ Gas Optimization / Suggestion	A vulnerability that has an informational character but is not affecting any of the code.

## Findings

Severity	Found
◆ Critical	0
◆ High-Risk	1
◆ Medium-Risk	0
◆ Low-Risk	2
◆ Gas Optimization / Suggestions	1



# MANUAL TESTING

---

**Centralization** – The owner can Blacklist Wallet.

**Severity: High**

**Function: blacklist**

**Status: Open**

**Overview:**

The owner can blacklist multiple wallets.

```
function blacklist(address _address, bool _isBlacklisting) external onlyOwner {  
    blacklists[_address] = _isBlacklisting;  
}
```

**Suggestion:**

There should be a locking period so that the wallet cannot be locked in an

indefinite

Period of time.



# MANUAL TESTING

---

## Centralization – Missing Events

**Severity:** Low

**Subject:** Missing Events

**Status:** Open

### Overview:

They serve as a mechanism for emitting and recording data onto the blockchain, making it transparent and easily accessible.

```
function blacklist(address _address, bool _isBlacklisting) external onlyOwner {  
    blacklists[_address] = _isBlacklisting;  
}  
function setRule(bool _limited, address _uniswapV2Pair, uint256  
_maxHoldingAmount, uint256 _minHoldingAmount) external onlyOwner {  
    limited = _limited;  
    uniswapV2Pair = _uniswapV2Pair;  
    maxHoldingAmount = _maxHoldingAmount;  
    minHoldingAmount = _minHoldingAmount;  
}
```

### Suggestion:

Emit an event for critical changes



# MANUAL TESTING

---

**Centralization – Missing zero/dead address.**

**Severity:** Low

**Status:** Open

**Subject:** Zero/dead address check.

**Overview:**

```
function setRule(bool _limited, address _uniswapV2Pair, uint256  
_maxHoldingAmount, uint256 _minHoldingAmount) external onlyOwner {  
    limited = _limited;  
    uniswapV2Pair = _uniswapV2Pair;  
    maxHoldingAmount = _maxHoldingAmount;  
    minHoldingAmount = _minHoldingAmount;  
}
```

**Suggestion:**

It is recommended that the address should not be zero or dead address check.



# MANUAL TESTING

---

## Optimization

**Severity:** Informational

**Subject:** Floating Pragma.

**Status:** Open

### Overview:

It is considered best practice to pick one compiler version and stick with it. With a floating pragma, contracts may accidentally be deployed using an outdated.

```
pragma solidity ^0.8.0;
```

### Suggestion:

Adding the latest constant version of solidity is recommended, as this prevents the unintentional deployment of a contract with an outdated compiler that contains unresolved bugs.



# DISCLAIMER

---

All the content provided in this document is for general information only and should not be used as financial advice or a reason to buy any investment. Team provides no guarantees against the sale of team tokens or the removal of liquidity by the project audited in this document. Always Do your own research and protect yourselves from being scammed. The Auditace team has audited this project for general information and only expresses their opinion based on similar projects and checks from popular diagnostic tools. Under no circumstances did Auditace receive a payment to manipulate those results or change the awarding badge that we will be adding in our website. Always Do your own research and protect yourselves from scams. This document should not be presented as a reason to buy or not buy any particular token. The Auditace team disclaims any liability for the resulting losses.



# ABOUT AUDITACE

---

We specialize in providing thorough and reliable audits for Web3 projects. With a team of experienced professionals, we use cutting-edge technology and rigorous methodologies to evaluate the security and integrity of blockchain systems. We are committed to helping our clients ensure the safety and transparency of their digital assets and transactions.



**<https://auditace.tech/>**



**[https://t.me/Audit\\_Ace](https://t.me/Audit_Ace)**



**[https://twitter.com/auditace\\_](https://twitter.com/auditace_)**



**<https://github.com/Audit-Ace>**

---