

## Security Assessment: Munch TOKEN





July 26, 2024

- Audit Status: **Pass**
- Audit Edition: **Advance**
































# Risk Analysis

## Classifications of Manual Risk Results

Classification	Description
 Critical	Danger or Potential Problems.
 High	Be Careful or Fail test.
 Medium	Pass, Not-Detected or Safe Item.
 Low	Function Detected

## Manual Code Review Risk Results

Contract Privilege	Description
 Buy Tax	3%
 Sale Tax	3%
 Cannot Buy	Pass
 Cannot Sale	Pass
 Max Tax	3%
 Modify Tax	Yes
 Fee Check	Pass
 Is Honeypot?	Not Detected
 Trading Cooldown	Not Detected
 Can Pause Trade?	Pass
 Pause Transfer?	Not-Detected
 Max Tx?	Fail
 Is Anti Whale?	Detected
 Is Anti Bot?	Not-Detected

Contract Privilege	Description
 Is Blacklist?	Not-Detected
 Blacklist Check	Pass
 is Whitelist?	Detected
 Can Mint?	Pass
 Is Proxy?	Not Detected
 Can Take Ownership?	Not Detected
 Hidden Owner?	Not-Detected
 Owner	0x22a1835B0c3e4A7cEF3310D87e94c1EA0c0d63f6
 Self Destruct?	Not Detected
 External Call?	Not-Detected
 Other?	Not Detected
 Holders	0
 Auditor Confidence	Low Risk
 KYC Present	No
 KYC URL	

The following quick summary it's added to the project overview; however, there are more details about the audit and its results. Please read every detail.

# Project Overview

## Token Summary

Parameter	Result
Address	0xeA2836C58B83d03c6F07Fc8035472DC1914f9F3b
Name	Munch
Token Tracker	Munch (MUNCH)
Decimals	18
Supply	1,000,000,000
Platform	BASE
compiler	0.8.19
Contract Name	MunchToken
Optimization	Yes with 200 runs
LicenseType	MIT
Language	Solidity
Codebase	<a href="https://basescan.org/address/0xea2836c58b83d03c6f07fc8035472dc1914f9f3b#code">https://basescan.org/ address/0xea2836c58b83d03c6f07fc8035472dc1914f9f3b#code</a>
Payment Tx	Corporate

## Main Contract Assessed Contract Name

Name	Contract	Live
Munch	0xeA2836C58B83d03c6F07Fc8035472DC1914f9F3b	Yes

## TestNet Contract Assessed Contract Name

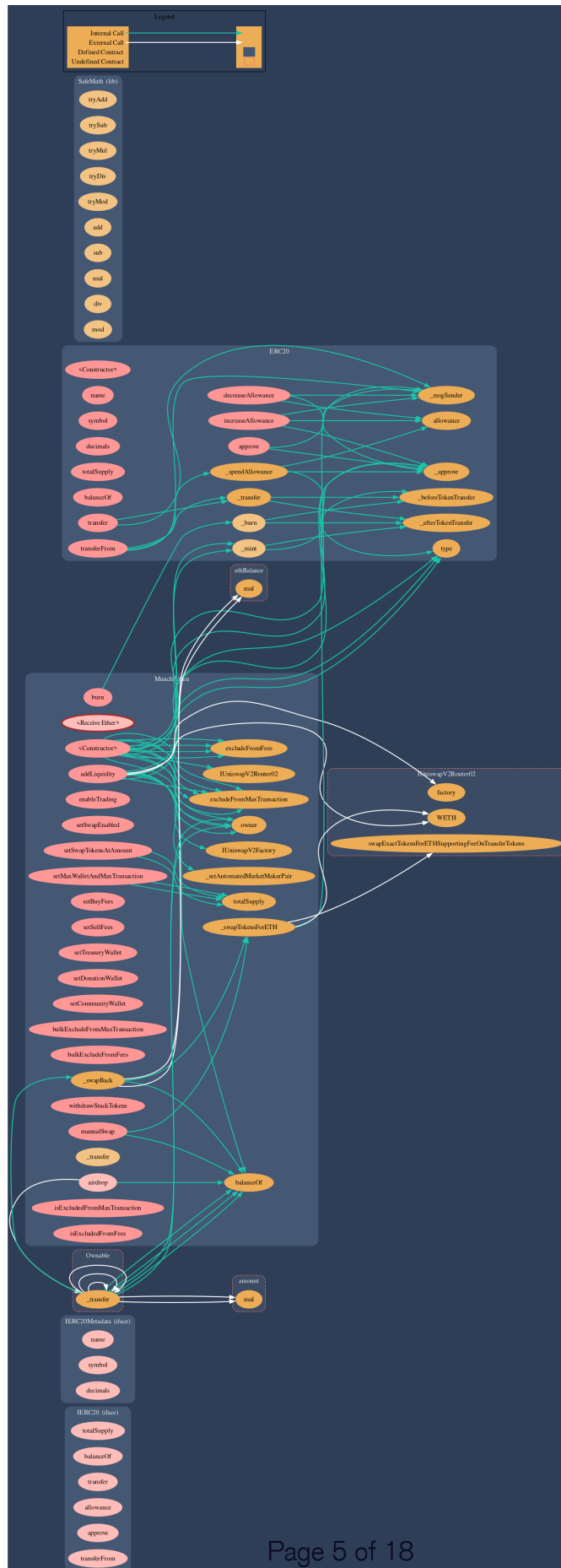
Name	Contract	Live
Munch	0xE44d49E61BA9Ee132BdB4035145Bc18cE2FE19f3	Yes

## Solidity Code Provided

SolidID	File Sha-1	FileName
Munch	9fad8344ef209460433179e36cfaf6473b7e4c9	MunchToken.sol
Munch		.sol
Munch		.sol
Munch		.sol
Munch		.sol
Munch		.sol

# Call Graph

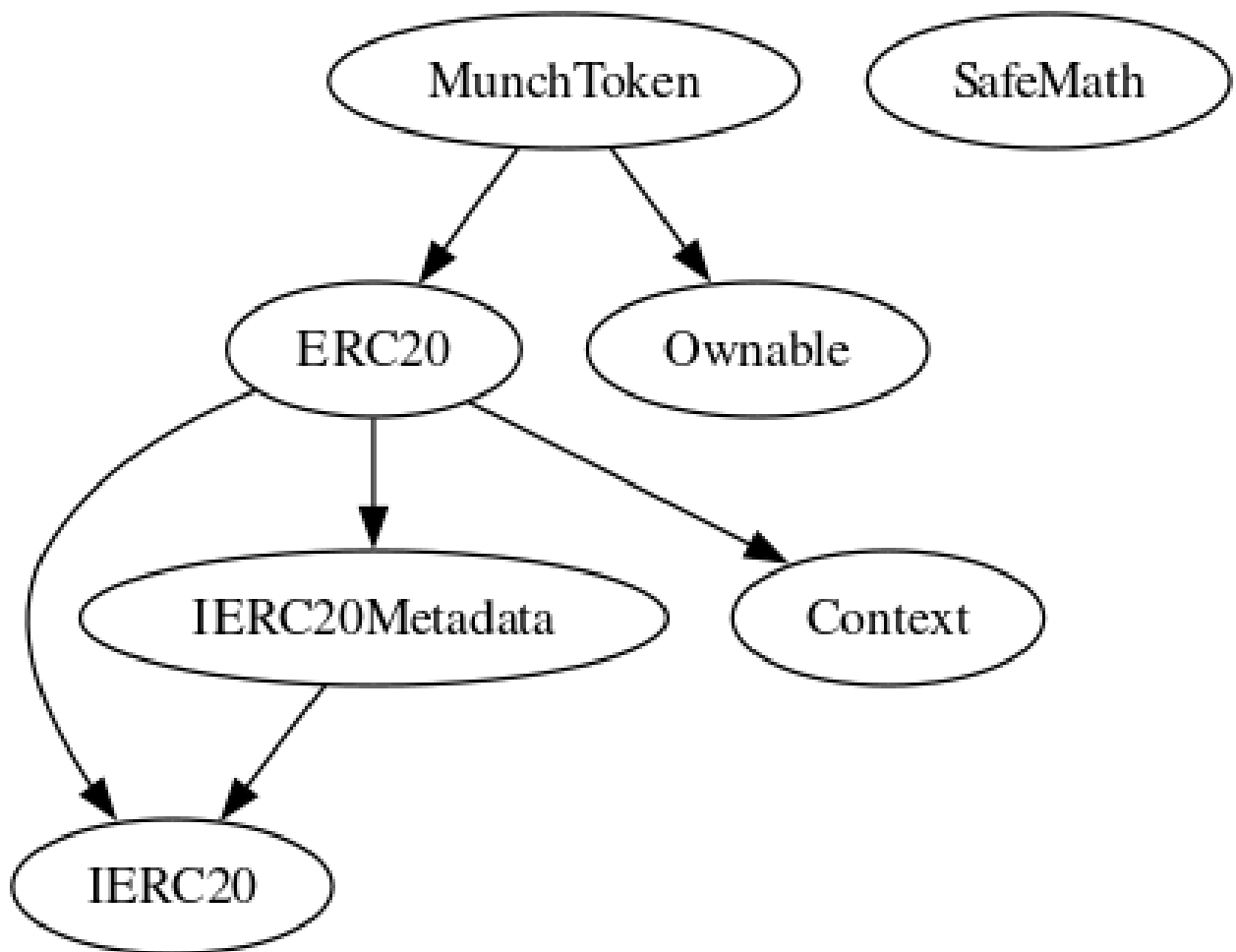
The contract for Munch has the following call graph structure.



# Inheritance

The contract for Munch has the following inheritance structure.

The Project has a Total Supply of 1,000,000,000



## Privileged Functions (onlyOwner)



Please Note if the contract is Renounced none of this functions can be executed.

Function Name	Parameters	Visibility
addLiquidity		Public
enableTrading		Public
setSwapEnabled	bool value	Public
setSwapTokensAtAmount	uint256 amount	Public
setMaxWalletAndMaxTransaction	uint256 _maxTransaction, uint256 _maxWallet	Public
setBuyFees	uint256 _treasuryFee, uint256 _donationFee, uint256 _communityFee	Public
setSellFees	uint256 _treasuryFee, uint256 _donationFee, uint256 _communityFee	Public
setTreasuryWallet	address _treasuryWallet	Public
setDonationWallet	address _donationWallet	Public



Function Name	Parameters	Visibility
setCommunityWallet	address _communityWallet	Public
excludeFromMaxTransaction	address account, bool value	Public
bulkExcludeFromMaxTransaction	address[] calldata accounts, bool value	Public
excludeFromFees	address account, bool value	Public
bulkExcludeFromFees	address[] calldata accounts, bool value	Public
manualSwap		Public
withdrawStuckTokens	address tkn	Public
airdrop	address[] calldata addresses, uint256[] calldata tokenAmounts	External

## MUNCH-01 | Potential Sandwich Attacks.

Category	Severity	Location	Status
Security	 Low	MunchToken.sol: L: 1198 C: 14	 Detected

### Description

A sandwich attack might happen when an attacker observes a transaction swapping tokens or adding liquidity without setting restrictions on slippage or minimum output amount. The attacker can manipulate the exchange rate by frontrunning (before the transaction being attacked) a transaction to purchase one of the assets and make profits by back running (after the transaction being attacked) a transaction to sell the asset. The following functions are called without setting restrictions on slippage or minimum output amount, so transactions triggering these functions are vulnerable to sandwich attacks, especially when the input amount is large:

- swapExactTokensForETHSupportingFeeOnTransferTokens()
- addLiquidityETH()



### Remediation

We recommend setting reasonable minimum output amounts, instead of 0, based on token prices when calling the aforementioned functions.

### References:

What Are Sandwich Attacks in DeFi — and How Can You Avoid Them?.

## MUNCH-02 | Function Visibility Optimization.

Category	Severity	Location	Status
Gas Optimization	 Informational	MunchToken.sol: L: 834 C: 14,L: 839 C: 14,L: 865 C: 14,L: 871 C: 14,L: 875 C: 14,L: 887 C: 14,L: 887 C: 14,L: 903 C: 14,L: 918 C: 14,L: 934 C: 14,L: 941 C: 14,L: 948 C: 14,L: 955 C: 14,L: 963 C: 14,L: 973 C: 14,L: 978 C: 14,L: 988 C: 14,L: 992 C: 14	 Detected

### Description

The following functions are declared as public and are not invoked in any of the contracts contained within the projects scope:

Function Name	Parameters	Visibility
burn	uint256 amount	Public
addLiquidity		Public
enableTrading		Public
setSwapEnabled	bool value	Public
setSwapTokensAtAmount	uint256 amount	Public
setMaxWalletAndMaxTransaction	uint256 _maxTransaction, uint256 _maxWallet	Public
setBuyFees	uint256 _treasuryFee, uint256 _donationFee, uint256 _communityFee	Public
setSellFees	uint256 _treasuryFee, uint256 _donationFee, uint256 _communityFee	Public
setTreasuryWallet	address _treasuryWallet	Public
setDonationWallet	address _donationWallet	Public

Function Name	Parameters	Visibility
setCommunityWallet	address _communityWallet	Public
excludeFromMaxTransaction	address account, bool value	Public
bulkExcludeFromMaxTransaction	address[] calldata accounts, bool value	Public
excludeFromFees	address account, bool value	Public
bulkExcludeFromFees	address[] calldata accounts, bool value	Public
manualSwap		Public
withdrawStuckTokens	address tkn	Public
airdrop	address[] calldata addresses, uint256[] calldata tokenAmounts	External
_setAutomatedMarketMakerPair	address pair, bool value	Internal
_swapBack		Internal
_swapTokensForETH	uint256 tokenAmount	Internal

The functions that are never called internally within the contract should have external visibility



## Remediation

We advise that the function's visibility specifiers are set to external, and the array-based arguments change their data location from memory to calldata, optimizing the gas cost of the function.

## References:

external vs public best practices.

## MUNCH-07 | State Variables could be Declared Constant.

Category	Severity	Location	Status
Coding Style	 Low	MunchToken.sol: L:720	 Not Detected

### Description

Constant state variables should be declared constant to save gas.

```
IUniswapV2Router02
```






### Remediation

Add the constant attribute to state variables that never changes.






<https://docs.soliditylang.org/en/latest/contracts.html#constant-state-variables>

# Technical Findings Summary

## Classification of Risk

Severity	Description
 Critical	Risks are those that impact the safe functioning of a platform and must be addressed before launch. Users should not invest in any project with outstanding critical risks.
 High	Risks can include centralization issues and logical errors. Under specific circumstances, these major risks can lead to loss of funds and/or control of the project.
 Medium	Risks may not pose a direct risk to users' funds, but they can affect the overall functioning of a platform
 Low	Risks can be any of the above but on a smaller scale. They generally do not compromise the overall integrity of the Project, but they may be less efficient than other solutions.
 Informational	Errors are often recommended to improve the code's style or certain operations to fall within industry best practices. They usually do not affect the overall functioning of the code.

## Findings

Severity	Found	Pending	Resolved
 Critical	0	0	1
 High	0	0	1
 Medium	0	0	3
 Low	2	2	3
 Informational	1	1	0
Total	3	3	8

# Social Media Checks

Social Media	URL	Result
Twitter	<a href="https://x.com/munchtoken">https://x.com/munchtoken</a>	Pass
Other		N/A
Website	<a href="https://munchproject.io">https://munchproject.io</a>	Pass
Telegram	<a href="https://t.me/MUNCHProjectportal">https://t.me/MUNCHProjectportal</a>	Pass

We recommend to have 3 or more social media sources including a completed working websites.

**Social Media Information Notes:**

**Auditor Notes:** undefined

**Project Owner Notes:**



# Audit Result

## Final Audit Score

Review	Score
Security Score	90
Auditor Score	90

The Following Score System Has been Added to this page to help understand the value of the audit, the maximum score is 100, however to attain that value the project must pass and provide all the data needed for the assessment. Our Passing Score has been changed to 85 Points for a higher standard, if a project does not attain 85% is an automatic failure. Read our notes and final assessment below.

## Audit Passed





# Assessment Results

## Important Notes:

- Overall Classification: I
- Performance: Low
- Score: 90/100

**Auditor Score =90**  
**Audit Passed**



# Appendix

## Finding Categories

### Centralization / Privilege

Centralization / Privilege findings refer to either feature logic or implementation of components that act against the nature of decentralization, such as explicit ownership or specialized access roles in combination with a mechanism to relocate funds.

### Gas Optimization

Gas Optimization findings do not affect the functionality of the code but generate different, more optimal EVM opcodes resulting in a reduction on the total gas cost of a transaction.

### Logical Issue

Logical Issue findings detail a fault in the logic of the linked code, such as an incorrect notion on how `block.timestamp` works.

### Control Flow

Control Flow findings concern the access control imposed on functions, such as owner-only functions being invoke-able by anyone under certain circumstances.

### Volatile Code

Volatile Code findings refer to segments of code that behave unexpectedly on certain edge cases that may result in a vulnerability.

### Coding Style

Coding Style findings usually do not affect the generated byte-code but rather comment on how to make the codebase more legible and, as a result, easily maintainable.

### Inconsistency

Inconsistency findings refer to functions that should seemingly behave similarly yet contain different code, such as a constructor assignment imposing different requirements on the input variables than a setter function.

### Coding Best Practices

ERC 20 Coding Standards are a set of rules that each developer should follow to ensure the code meets a set of criteria and is readable by all the developers.

# Disclaimer

Assure Defi has conducted an independent security assessment to verify the integrity of and highlight any vulnerabilities or errors, intentional or unintentional, that may be present in the reviewed code for the scope of this assessment. This report does not constitute agreement, acceptance, or advocacy for the Project, and users relying on this report should not consider this as having any merit for financial advice in any shape, form, or nature. The contracts audited do not account for any economic developments that the Project in question may pursue, and the veracity of the findings thus presented in this report relate solely to the proficiency, competence, aptitude, and discretion of our independent auditors, who make no guarantees nor assurance that the contracts are entirely free of exploits, bugs, vulnerabilities or deprecation of technologies.

All information provided in this report does not constitute financial or investment advice, nor should it be used to signal that any persons reading this report should invest their funds without sufficient individual due diligence, regardless of the findings presented. Information is provided 'as is, and Assure Defi is under no covenant to audited completeness, accuracy, or solidity of the contracts. In no event will Assure Defi or its partners, employees, agents, or parties related to the provision of this audit report be liable to any parties for, or lack thereof, decisions or actions with regards to the information provided in this audit report.

The assessment services provided by Assure Defi are subject to dependencies and are under continuing development. You agree that your access 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 with high levels of technical risk and uncertainty. The assessment reports could include false positives, negatives, and unpredictable results. The services may access, and depend upon, multiple layers of third parties.

