QuillAudits

AUDIT REPORT

July 2025

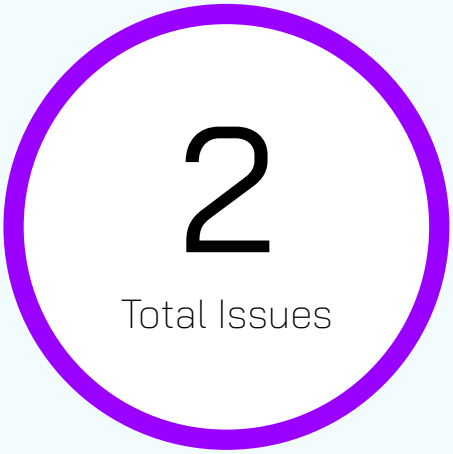
For

Table of Content

Executive Summary

Project name	Swarm
Project URL	https://swarm.com/
Overview	Give codebase is Move smart contract for the Swarm Network Token (MRAWS), a cryptocurrency token built on the Sui blockchain. Symbol: MRAWS Name: The Mraws Token Decimals: 8 Total Supply: 10,000,000,000 (10 billion tokens)
Audit Scope	The scope of this Audit was to analyze the Swarm Token Smart Contracts for quality, security, and correctness.
Source Code link	https://github.com/Swarm-Network-work/swarm-token-SUI/blob/main/sources/swarm_network_token.move
Contracts in Scope	swarm_network_token.move
Commit Hash	7fbc4ccfa1b6700de2c2c06053efb2fa105d0852
Language	Move
Blockchain	Sui
Method	Manual Analysis, Functional Testing, Automated Testing
Updated Code Received	23rd July 2025
Review 2	23rd July 2025
Fixed In	None

Number of Issues per Severity



High	0 (0.00%)
Medium	0 (0.00%)
Low	0 (0.00%)
Informational	2 (100.00%)

Severity

Issues

	High	Medium	Low	Informational
Open	0	0	0	0
Resolved	0	0	0	0
Acknowledged	0	0	0	**2**
Partially Resolved	0	0	0	0

Checked Vulnerabilities

- ☑ Access Management
- ☑ Transaction-ordering dependence
- ☑ Timestamp dependence
- ☑ Integer overflow/underflow by bit operations
- ☑ Number of rounding errors
- ☑ Denial of service / logical oversights
- ☑ Access control
- ☑ Centralization of power
- ☑ Business logic contradicting the specification
- ☑ Code clones
- ☑ functionality duplication
- ☑ Gas usage
- ☑ Arbitrary token minting
- ☑ Unchecked CALL Return Values
- ☑ The flow of capability
- ☑ Witness Type
- ☑ Implicit visibility level

Techniques and Methods

Throughout the audit of smart contracts, care was taken to ensure:

- The overall quality of code
- Use of best practices
- Code documentation and comments, match logic and expected behavior
- Token distribution and calculations are as per the intended behavior mentioned in the whitepaper
- Efficient use of gas
- Code is safe from re-entrancy and other vulnerabilities

The following techniques, methods, and tools were used to review all the smart contracts:

Structural Analysis

In this step, we have analyzed the design patterns and structure of smart contracts. A thorough check was done to ensure the smart contract is structured in a way that will not result in future problems.

Static Analysis

A static Analysis of Smart Contracts was done to identify contract vulnerabilities. In this step, a series of automated tools are used to test the security of smart contracts.

Code Review / Manual Analysis

Manual Analysis or review of code was done to identify new vulnerabilities or verify the vulnerabilities found during the static analysis. Contracts were completely manually analysed, and their logic was checked and compared with the one described in the whitepaper. Besides, the results of the automated analysis were manually verified.

Gas Consumption

In this step, we have checked the behaviour of Solana programs in production. Checks were done to know how much gas gets consumed and the possibilities of optimising code to reduce gas consumption.

Types of Severity

Every issue in this report has been assigned to a severity level. There are four levels of severity, and each of them has been explained below

High Severity Issues

A high severity issue or vulnerability means that your smart contract can be exploited. Issues on this level are critical to the smart contract's performance or functionality, and we recommend these issues be fixed before moving to a live environment.

Medium Severity Issues

The issues marked as medium severity usually arise because of errors and deficiencies in the smart contract code. Issues on this level could potentially bring problems, and they should still be fixed.

Low Severity Issues

Low-level severity issues can cause minor impact and are just warnings that can remain unfixed for now. It would be better to fix these issues at some point in the future.

Informational Issues

These are four severity issues that indicate an improvement request, a general question, a cosmetic or documentation error, or a request for information. There is low-to-no impact.

Types of Issues

Open

Security vulnerabilities identified that must be resolved and are currently unresolved.

Resolved

Security vulnerabilities identified that must be resolved and are currently unresolved.

Acknowledged

Vulnerabilities which have been acknowledged but are yet to be resolved.

Partially Resolved

Considerable efforts have been invested to reduce the risk/ impact of the security issue, but are not completely resolved.

Informational Severity Issues

One-Time Mint Vulnerability

Acknowledged

Path

sources/swarm_network_token.move

Function

mint

Description

Mint function can fail, leaving tokens permanently unmintable

```
public entry fun mint(
treasury_cap: &mut TreasuryCap<SWARM_NETWORK_TOKEN>,
mint_cap: MintCap,
ctx: &mut TxContext,
) {
// Consuming the mint_cap ensures this can only be called once
let MintCap { id } = mint_cap;
object::delete(id);
coin::mint_and_transfer(treasury_cap, TOTAL_SUPPLY, TREASURY_WALLET, ctx)
}
```

Impact:

- If mint transaction fails after MintCap is consumed, tokens cannot be minted again
- No recovery mechanism for failed mint operations
- Could result in zero token supply if mint fails

Recommendation

Confirm if it's intended design and ensure adequate error handling are implemented

Swarm Team's comment :

It is intended indeed since we will check on deploy which will be manual inspection since it will be called once anyways.

Token Symbol Mismatch

Acknowledged

Path

sources/swarm_network_token.move

Function

-

Description

Branding confusion,Token symbol and name is not matching as mentioned in “Read.me”

- Symbol "TRUTH" doesn't align with project name
- Could confuse users and exchanges

```
let (treasury, metadata) = coin::create_currency(
witness,
8,
b"TRUTH",
b"Swarm Network",
```

Swarm Team's comment :

The ticker is indeed $TRUTH

Automated Tests

No major issues were found. Some false positive errors were reported by the tools. All the other issues have been categorized above according to their level of severity.

Closing Summary

In this report, we have considered the security of Swarm. We performed our audit according to the procedure described above.

Informational severity issues were found. Swarm team acknowledged all the issues mentioned

Disclaimer

At QuillAudits, we have spent years helping projects strengthen their smart contract security. However, security is not a one-time event—threats evolve, and so do attack vectors. Our audit provides a security assessment based on the best industry practices at the time of review, identifying known vulnerabilities in the received smart contract source code.

This report does not serve as a security guarantee, investment advice, or an endorsement of any platform. It reflects our findings based on the provided code at the time of analysis and may no longer be relevant after any modifications. The presence of an audit does not imply that the contract is free of vulnerabilities or fully secure.

While we have conducted a thorough review, security is an ongoing process. We strongly recommend multiple independent audits, continuous monitoring, and a public bug bounty program to enhance resilience against emerging threats.

Stay proactive. Stay secure.

About QuillAudits

QuillAudits is a leading name in Web3 security, offering top-notch solutions to safeguard projects across DeFi, GameFi, NFT gaming, and all blockchain layers. With seven years of expertise, we've secured over 1400 projects globally, averting over $3 billion in losses. Our specialists rigorously audit smart contracts and ensure DApp safety on major platforms like Ethereum, BSC, Arbitrum, Algorand, Tron, Polygon, Polkadot, Fantom, NEAR, Solana, and others, guaranteeing your project's security with cutting-edge practices.

7+ Years of Expertise	**1M+** Lines of Code Audited
50+ Chains Supported	**1400+** Projects Secured

Follow Our Journey

AUDIT REPORT

July 2025

For

QuillAudits

Canada, India, Singapore, UAE, UK

www.quillaudits.com audits@quillaudits.com