



# AUDIT REPORT

---

January, 2025

For

 BNB Agents

# Table of Content

Executive Summary	03
Number of Severity Per Issues	05
Checked Vulnerabilities	06
Techniques & Methods	07
Types of Severity	09
Types of Issues	10
Closing Summary & Disclaimer	11

# Executive Summary

**Project name** BNBAi

**Overview** BNBAi is a digital asset leveraging the ERC20 standard for token creation and management on the blockchain. It introduces a vast supply of 1 Billion tokens, aiming to serve as a utility or governance token within its ecosystem. Unique features include EIP-2612 permit functionality, allowing token spending without gas fees through signed messages, and advanced transfer mechanisms optimized for efficiency. BNBAi stands out with its infinite allowance option for specific contracts, enhancing interoperability within DeFi protocols.

**Project URL** <https://www.bnbagents.ai/>

**Audit Scope** The scope of this audit was to analyse the BNBAi Token Contract for quality, security, and correctness.

**Source Code** <https://bsc-scan.com/token/0x2aabe2ef9ee8ab04c6f27c4284c3f268769b35ec>

**Method** Manual Review, Functional Testing, Automated Testing, etc. All the raised flags were manually reviewed and re-tested to identify any false positives.

**Blockchain** BSC

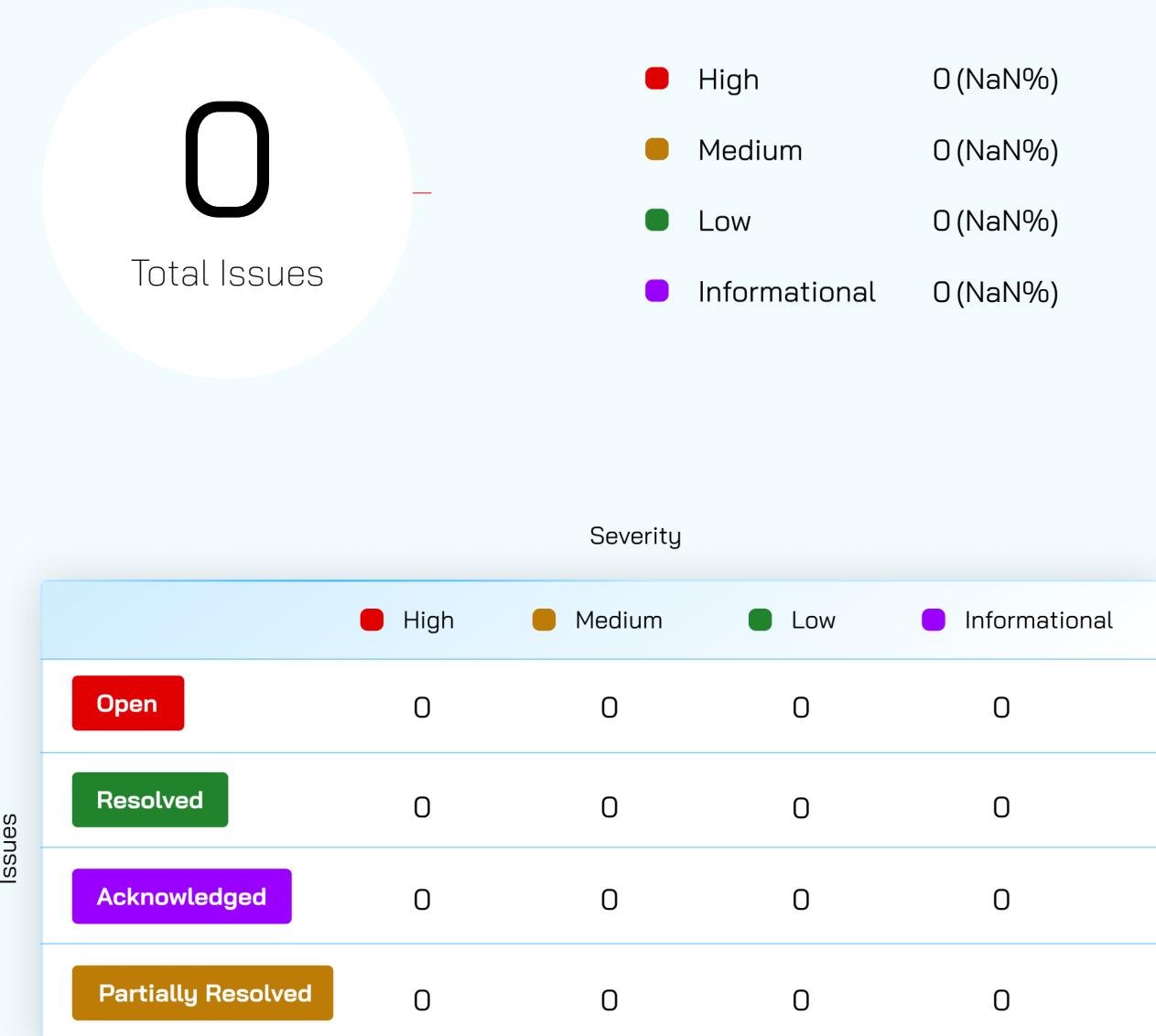
**Review 1** 29th January 2025

**Updated Code Received** NA

**Review 2** NA

**Fixed In** NA

# Number of Issues per Severity



# Checked Vulnerabilities

<input checked="" type="checkbox"/> Re-entrancy	<input checked="" type="checkbox"/> Implicit Visibility Level
<input checked="" type="checkbox"/> Timestamp Dependence	<input checked="" type="checkbox"/> Access Management
<input checked="" type="checkbox"/> Gas Limit and Loops	<input checked="" type="checkbox"/> Centralization of Control
<input checked="" type="checkbox"/> DoS with Block Gas Limit	<input checked="" type="checkbox"/> Unsafe Type Inference
<input checked="" type="checkbox"/> Transaction-Ordering Dependence	<input checked="" type="checkbox"/> Style Guide Violation
<input checked="" type="checkbox"/> Exception Disorder	<input checked="" type="checkbox"/> Logical Issues and Flaws
<input checked="" type="checkbox"/> Balance Equality	<input checked="" type="checkbox"/> Arithmetic Computations Correctness
<input checked="" type="checkbox"/> Transfer Forwards All Gas	<input checked="" type="checkbox"/> Malicious Libraries
<input checked="" type="checkbox"/> Compiler Version Not Fixed	<input checked="" type="checkbox"/> Revert/Require Functions
<input checked="" type="checkbox"/> Redundant Fallback Function	<input checked="" type="checkbox"/> Private Modifier
<input checked="" type="checkbox"/> Send Instead of Transfer	<input checked="" type="checkbox"/> Missing Zero Address Validation
<input checked="" type="checkbox"/> Unchecked External Call	<input checked="" type="checkbox"/> Return Values of Low-Level Calls
<input checked="" type="checkbox"/> Unchecked Math	

# 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.
- Implementation of ERC standards.
- 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 analyzed, 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 behavior of smart contracts in production. Checks were done to know how much gas gets consumed and the possibilities of optimization of code to reduce gas consumption.

**Tools And Platforms Used For Audit**

Remix IDE, Foundry, Solhint, Mythril, Slither, Solidity statistical analysis.

# 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

<b>Open</b>  Security vulnerabilities identified that must be resolved and are currently unresolved.	<b>Resolved</b>  Security vulnerabilities identified that must be resolved and are currently unresolved.
<b>Acknowledged</b>  Vulnerabilities which have been acknowledged but are yet to be resolved.	<b>Partially Resolved</b>  Considerable efforts have been invested to reduce the risk/ impact of the security issue, but are not completely resolved.

# No Issues Found

# 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 the BNBAi Token contract. We performed our audit according to the procedure described above.

Code looks Good, No Issues Found.

# Disclaimer

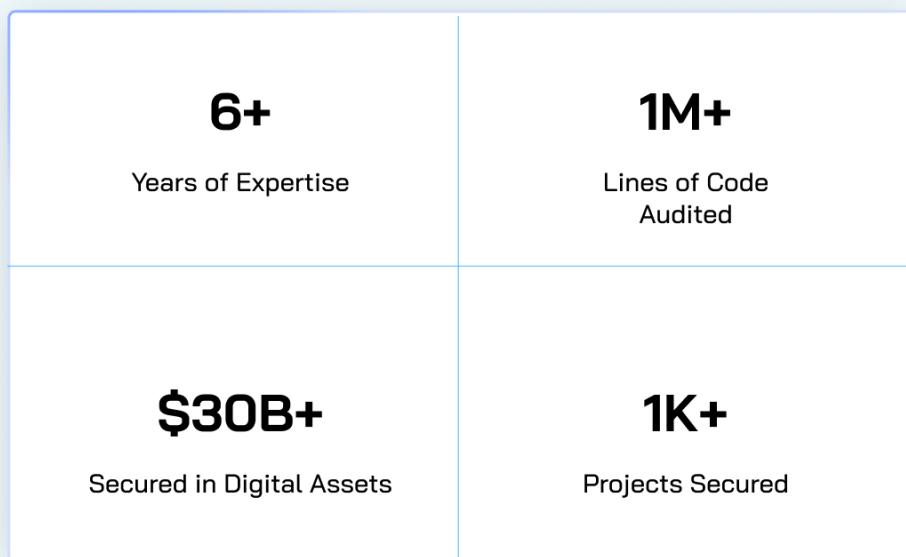
QuillAudits Smart contract security audit provides services to help identify and mitigate potential security risks in BNBAi smart contract. However, it is important to understand that no security audit can guarantee complete protection against all possible security threats. QuillAudits audit reports are based on the information provided to us at the time of the audit, and we cannot guarantee the accuracy or completeness of this information. Additionally, the security landscape is constantly evolving, and new security threats may emerge after the audit has been completed.

Therefore, it is recommended that multiple audits and bug bounty programs be conducted to ensure the ongoing security of BNBAi smart contract. One audit is not enough to guarantee complete protection against all possible security threats. It is important to implement proper risk management strategies and stay vigilant in monitoring your smart contracts for potential security risks.

QuillAudits cannot be held liable for any security breaches or losses that may occur subsequent to and despite using our audit services. It is the responsibility of the BNBAi to implement the recommendations provided in our audit reports and to take appropriate steps to mitigate potential security risks.

# 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 six years of expertise, we've secured over 1000 projects globally, averting over \$30 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.



Follow Our Journey



# AUDIT REPORT

---

January, 2025

For

 **BNB Agents**



Canada, India, Singapore, UAE, UK

[www.quillaudits.com](http://www.quillaudits.com)    [audits@quillaudits.com](mailto:audits@quillaudits.com)