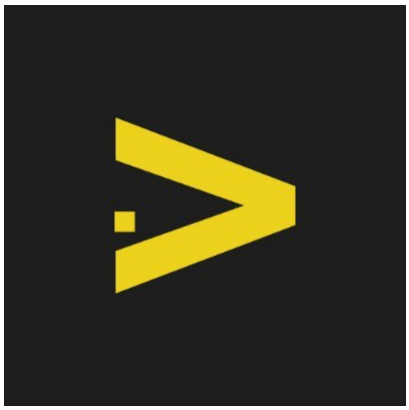
verichains

SECURITY AUDIT OF

BIGINT TOKEN



Public Report

Jul 1, 2024

Verichains Lab

info@verichains.io

https://www.verichains.io

Driving Technology > Forward

ABBREVIATIONS

Name	Description
Ethereum	An open source platform based on blockchain technology to create and distribute smart contracts and decentralized applications.
Ether (ETH)	A cryptocurrency whose blockchain is generated by the Ethereum platform. Ether is used for payment of transactions and computing services in the Ethereum network.
Smart contract	A computer protocol intended to digitally facilitate, verify or enforce the negotiation or performance of a contract.
Solidity	A contract-oriented, high-level language for implementing smart contracts for the Ethereum platform.
Solc	A compiler for Solidity.
ERC20	ERC20 (BEP20 in Binance Smart Chain or *x*RP20 in other chains) tokens are blockchain-based assets that have value and can be sent and received. The primary difference with the primary coin is that instead of running on their own blockchain, ERC20 tokens are issued on a network that supports smart contracts such as Ethereum or Binance Smart Chain.

EXECUTIVE SUMMARY

This Security Audit Report was prepared by Verichains Lab on Jul 1, 2024. We would like to thank the BigInt for trusting Verichains Lab in auditing smart contracts. Delivering high-quality audits is always our top priority.

This audit focused on identifying security flaws in code and the design of the BIGINT Token. The scope of the audit is limited to the source code files provided to Verichains. Verichains Lab completed the assessment using manual, static, and dynamic analysis techniques.

During the audit process, the audit team had identified no vulnerable issue in the smart contracts code.

TABLE OF CONTENTS

1. MANAGEMENT SUMMARY

1.1. About BIGINT Token

BIGINT Token is a governance token used in the BigInt ecosystem, which is an NFT and MEME incubator focused on digital art and meme tokens, with a heavy emphasis on community interaction. Tokens have the role of providing liquidity and rewarding users in the ecosystem.

1.2. Audit scope

This audit focused on identifying security flaws in code and the design of the BIGINT Token.

The latest version of the following files were made available in the course of the review:

SHA256 Sum	File
30202f929513f5e866a2fe011a95dd9cc656c38827b382daca509973d3a8a237	BigintToken.sol

1.3. Audit methodology

Our security audit process for smart contract includes two steps:

- Smart contract codes are scanned/tested for commonly known and more specific vulnerabilities using public and RK87, our in-house smart contract security analysis tool.
- Manual audit of the codes for security issues. The contracts are manually analyzed to look for any potential problems.

Following is the list of commonly known vulnerabilities that were considered during the audit of the smart contract:

- Integer Overflow and Underflow
- Timestamp Dependence
- Race Conditions
- Transaction-Ordering Dependence
- DoS with (Unexpected) revert
- DoS with Block Gas Limit
- Gas Usage, Gas Limit and Loops
- Redundant fallback function
- Unsafe type Inference
- Reentrancy
- Explicit visibility of functions state variables (external, internal, private and public)

- Logic Flaws

For vulnerabilities, we categorize the findings into categories as listed in table below, depending on their severity level:

SEVERITY LEVEL	DESCRIPTION
CRITICAL	A vulnerability that can disrupt the contract functioning; creates a critical risk to the contract; required to be fixed immediately.
HIGH	A vulnerability that could affect the desired outcome of executing the contract with high impact; needs to be fixed with high priority.
MEDIUM	A vulnerability that could affect the desired outcome of executing the contract with medium impact in a specific scenario; needs to be fixed.
LOW	An issue that does not have a significant impact, can be considered as less important.

Table 1. Severity levels

1.4. Disclaimer

BigInt acknowledges that the security services provided by Verichains, are conducted to the best of their professional abilities but cannot guarantee 100% coverage of all security vulnerabilities. BigInt understands and accepts that despite rigorous auditing, certain vulnerabilities may remain undetected. Therefore, BigInt agrees that Verichains shall not be held responsible or liable, and shall not be charged for any hacking incidents that occur due to security vulnerabilities not identified during the audit process.

1.5. Acceptance Minute

This final report served by Verichains to the BigInt will be considered an Acceptance Minute. Within 7 days, if no any further responses or reports is received from the BigInt, the final report will be considered fully accepted by the BigInt without the signature.

2. AUDIT RESULT

2.1. Overview

The BIGINT Token was written in `Solidity` language, with the required version to be `0.8.0`.

The contract extends `ERC20Burnable` and `Ownable` from the `OpenZeppelin` library. Below table describes some properties of the audited BIGINT Token (as of the report writing time).

PROPERTY	VALUE
Name	Bigint
Symbol	BIGINT
Decimals	18
Total Supply	1,050,000,000x10^{18} (it represents one billion and fifty million)

Table 2. The BIGINT Token properties

It includes a whitelist mechanism, allowing the owner to grant or revoke special transfer permissions for specific addresses. Additionally, the owner can set a liquidity pair address. Transfers are restricted to the owner and whitelisted addresses until the liquidity pair is set, ensuring controlled trading. Specially, the owner can transfer or renounce ownership.

2.2. Findings

During the audit process, the audit team found no vulnerability in the given version of BIGINT Token.

APPENDIX

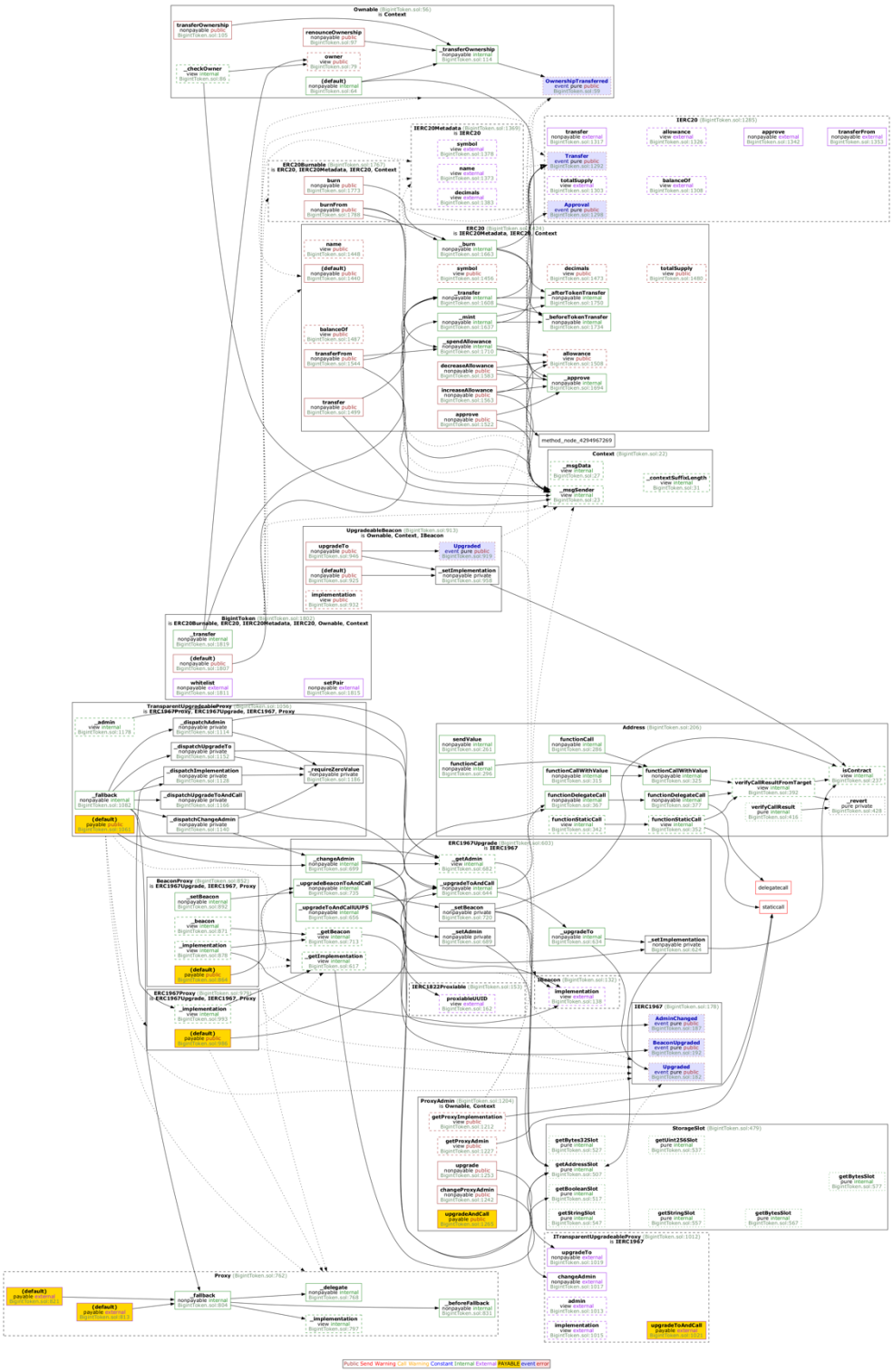


Image 1. ABI token smart contract call graph

3. VERSION HISTORY

Version	Date	Status/Change	Created by
1.0	*Jul 1, 2024*	Public Report	Verichains Lab

Table 3. Report versions history